NR INDEKSU 353965
PL ISSN 0860-1674

1

Z MIKROKOMPUTEREM NA TY

Bajtek

MIESIĘCZNY DODATEK DO SZTANDARU MŁODYCH NR 1(25) STYCZEŃ 1988 CENA 100 ZŁ

TALERZEM W NIEBO

BITIMAGE

KREOWANIE BYTÓW

BALTCOM'87

SPIS TREŚCI BAJTKA 1986/1987

SM SZTANDAR MŁODYCH

UL WRITEST

MIKROKOMPUTEROWY MODEL RODZINY PSZCZELEJ

TECHNO-ETYKA

Jakie konsekwencje dla przyszłości człowieka może przynieść burzliwy rozwój komputeryzacji?

Wiadomo że postęp naukowo-techniczny otwiera przed gatunkiem homo sapiens nowe perspektywy. Człowiek będzie w coraz większym stopniu uwalniany od pracy ciężkiej, niebezpiecznej. Coraz częściej nasza praca będzie miała charakter twórczy.

Ale występuje również ciekawa relacja odwrotna. Technika komputerowa wymaga od człowieka odpowiedniej kultury posługiwania się nią. Profesor Iwan Frołow, radziecki filozof zajmujący się konsekwencjami rozwoju nauki i techniki, uogólnia to stawiając tezę, że nowoczesnej, awangardowej technologii powinien odpowiadać nowy, wyższy poziom rozwoju samego człowieka, nie tylko jego cech zawodowych, ale i ogólnokulturowych, w tym i umiejętności obcowania z otaczającą przyrodą.

Profesor Frołow wprowadził na określenie tych wymagań termin „wysoka styczność". Właśnie wymóg „wysokiej styczności" jest, jego zdaniem, warunkiem przetrwania i rozwoju człowieka i społeczeństwa. Techniką zajmowaliśmy się już dostatecznie dużo. Teraz, aby przejść na wyższy etap rozwoju cywilizacyjnego, w większym stopniu powinniśmy zająć się bezpośrednio wszechstronnym rozwojem samego człowieka.

Nie jest to bynajmniej problem li tylko teoretyczny. Nieprzygotowany człowiek może stać się hamulcem postępu, może powstrzymać rozwój. Dlatego już obecnie inżynierowie konstruujący złożone systemy techniczne zastanawiają się nie tylko nad tym, jak uczynić je bardziej efektywnymi, jak je zoptymalizować, ale również nad tym, jak... uchronić je przed niekompetentnym i nieprzygotowanym człowiekiem. Katastrofa w Czernobylu — wywołana błędnymi działaniami pracujących tam inżynierów — najdobitniej wykazała, że nie jest to bynajmniej problem teoretyczny!

Uwzględniając omylność człowieka można by, oczywiście, próbować tak programować rozwój, aby podejmowanie zasadniczych decyzji pozostawić „niezawodnym" maszynom. Tylko że z wielu powodów była by to droga prowadząca donikąd. Ciekawe spojrzenie w tej kwestii na zależności między człowiekiem a maszyną proponuje radziecki socjolog i futurolog profesor Igor Bestużew-Łada. Twierdzi on, że jeśli człowiek podejdzie bezkrytycznie do komputeryzacji, to może znaleźć się w sytuacji głupiego pana uzależnionego od mądrego sługi.

Profesor Bestużew-Łada lubi powoływać się na nowelę science fiction o Romeo i Julii XXI wieku. Ale ta Julia nie chce Romea, znudził się jej. On robi wszystko, żeby ją odzyskać, wreszcie sięga po pomoc swojego komputera. I już wydaje się, że Julia nie ma innego wyjścia, niż ulec — przecież komputer przeanalizował wszystkie warianty... W ostatnim jednak momencie, gdy już komputer zaczyna wyświetlać na ekranie właściwą podpowiedź, Romeo uświadamia sobie, że jest człowiekiem, homo sapiens, i popełnia samobójstwo. I, w pewnym sensie, można uznać, że czyni to słusznie, gdyż w tym samym momencie Julia gorączkowo pyta swój komputer, jak uchronić się przed natarczywym Romeo...

Wynika z tego przykładu, że jeżeli powierzymy wszystko komputerom, to znajdziemy się w odczłowieczonym, nieludzkim społeczeństwie. W ciągu ostatnich kilkuset tysięcy lat wykształciła się taka forma społecznego współżycia, którą nazywamy etyką. Są to zasady uczciwego, właściwego zachowania się. Profesor Bestużew-Łada uważa, że obecnie jedna etyka to za mało. Potrzebne są trzy etyki: etyka stosunków między człowiekiem a maszyną przyrodą, oraz techno-etyka, czyli etyka stosunków między człowiekiem a maszyną.

Musi stać się jasne, że w tej sprawie można zdać się na komputer, a w tej nie, bo byłoby wstyd, tak jak wstyd jest wyjść bez spodni na ulicę. Nie wypada pytać się komputera, jak postępować z Julią, tak jak nie wypada rzucać się na ulicy do całowania obcej kobiety. Dopiero te trzy etyki mogą, zdaniem radzieckiego futurologa, uratować homo sapiens w XXI wieku.

Jaki więc główny wniosek wynika dla człowieka analizującego burzliwy rozwój informatyki? Taki przede wszystkim, że dysponując coraz bardziej doskonałym komputerem, człowiek musi starać się być coraz wszechstronniej wykształcony i rozwinięty. Może warto więc Młody Przyjacielu wyłączyć na parę godzin Twoją wspaniałą „maszynę" i przeczytać na przykład „O tym, co Alicja odkryła po Drugiej Stronie Lustra"?! Zwłaszcza, że i w tej pięknej opowieści kryją się matematyczne tajemnice.

Waldemar Siwiński



SZANOWNY PANIE REDAKTORZE!

Udało mi się zdobyć jeden numer Waszego pisma i jestem zachwycony jego poziomem i aktualnością doniesień. W przyszłym roku mam zamiar zaprenumerować Wasze pismo żeby na bieżąco śledzić rozwój domowej informatyki w Polsce. Wiele tematów poruszanych w Waszym piśmie jest tutaj pomijanych. Nie mówiąc już o tym, że o polskich czcionkach w popularnych magazynach nie pisze się w ogóle.

Posiadam IBM PC XT kompatybilny komputer, z dwoma stacjami dysków i pamięcią 640 KB, drukarkę Copal Write Hand 1200 L oraz myszkę LOGIMOUSE C7. Jestem zainteresowany wymianą doświadczeń, tricków programowych jak i samych programów. Programuję w Quick Basic v.3.0, bardzo mało w assemblerze i próbuję programować w Turbo Pascal-u. Ostatnia wersja Quick Basic-a jest jednak tak wszechstronna i kompilowane programy tak szybkie, że rzadko widzę potrzebę zastosowania innego języka. Posiadam wiele programów komercyjnych (w zasadzie większość standardów jak Wordperfect v.4.2, dBase III+, LOTUS 123 v.2.0, Symphony itp.) oraz wiele tak zwanych programów public domain.

Byłbym bardzo wdzięczny, gdyby Wesze pismo mogło wydrukować krótką notatkę z moim adresem i z informacją, że chciałbym nawiązać kontakt z kolegami mającymi dostęp lub posiadającymi IBM kompatybilny komputer.

Życzę Waszemu pismu dalej tak burzliwego rozwoju i trzymania dobrego poziomu. Z poważaniem i pozdrowieniami dla całej redakcji i wszystkich czytelników.

Andrzej Kajetanowicz
1263 Queen St. Apt. 516
Halifax, Nova Scotia
B3J 3L4
CANADA

Czytam Wasze pismo od pierwszego jeszcze zeszytowego numeru. Uważam, ze robicie wspaniałą robotę, pozwalając ludziom takim jak ja, ludziom nie mającym własnego sprzętu, marzyć o nim i przygotowujecie nas na tę chwilę, gdy już dostaniemy komputer „w swoje łapy". To naprawdę fajnie, ze po prostu jesteście. Dzięki Wam mam nadzieję zrobić dobry użytek z komputera w przyszłości. Moja do Was prośba — przedstawiajcie jak najwięcej zastosowań. Mnie interesuje włączenie komputera do systemu pomiarowego wielkości elektrycznych oraz wykorzystanie jako inteligentnego sterownika urządzeń domowych. W jaki sposób i przy pomocy jakich elementów dodatkowych można to osiągnąć. I jeszcze to, co pojawiło się w pierwszych waszych numerach — usprawnienia i przeróbki dla średnio zaawansowanych elektroników — mógłby z nich powstać stały kącik „komputerowego majsterklepki".

Dziś biorę udział w Waszym Świątecznym Konkursie i marzę by to właśnie moją odpowiedź wylosował Kubuś Literka.

Z noworocznymi życzeniami podwajania nakładu co kwartał i nadzieją na łut szczęścia w konkursie

Zawsze Wam życzliwy pozostaję.
Jarosław Raj

WYBIERZ SAM

„BAJTEK" — MIESIĘCZNY DODATEK DO „SZTANDARU MŁODYCH"

ADRES: 00-687 Warszawa, ul. Wspólna 61. Tel. 21-12-05
Przewodniczący Rady Redakcyjnej: Jerzy Domański - redaktor naczelny „Sztandaru Młodych".

ZESPÓŁ REDAKCYJNY: Waldemar Siwiński (z-ca redaktora naczelnego „SM" — kierownik zespołu „Bajtka"), Roman Poznański (z-ca sekretarza redakcji „SM" — sekretarz zespołu „Bajtka"), Krzysztof Czernek, Sławomir Gajda (red. techniczny), Andrzej Gogolewski, Andrzej Kowalewski, Andrzej Podulka, Sławomir Polak, Wanda Roszkowska (opr. graficzne), Kazimierz Treger, Marcin Waligórski, Roman Wojciechowski. Zdjęcia w numerze. Leopold Dzikowski.

Klany redagują:
Commodore — Klaudiusz Dybowski,
Amstrad-Schneider — Tomasz Pyć, Sergiusz Wolicki,
Spectrum — Marcin Przasnyski, Michał Szuniewicz,
Atari — Wiesław Migut, Wojciech Zientara.

Fotoskład — Tadeusz Olczak,
Montaż offsetowy — Grażyna Ostaszewska,
Korekta — Maria Krajewska, Zofia Wółtańska

WYDAWCA: RSW „Prasa-Książka-Ruch" Młodzieżowa Agencja Wydawnicza, al. Stanów Zjednoczonych 53, 04-028 Warszawa. Telefony: Centrala 13-20-40 do 49, Redakcja Reklamy 13-20-40 do 49 w. 403, 414
Cena 100 zł.
Skład techniką CRT-200, przygotowalnia offsetowa i druk PRASOWE ZAKŁADY GRAFICZNE RSW „PRASA-KSIĄŻKA-RUCH" w Ciechanowie, ul Sienkiewicza 51.
Nr zlecenia 138827 n. 200 000 egz. U-113



Bajtek



JESTEŚMY ZBUDOWANI Z TEGO, CO PRZYRODA MIAŁA POD RĘKĄ

KREOWANIE BYTÓW

Rozmowa z prof. drem hab. inż. Ryszardem Tadeusiewiczem, kierownikiem Zakładu Biocybernetyki Akademii Górniczo-Hutniczej w Krakowie.

— Cybernetyka od początku związana była ściśle z biologią. Czym więc w takim razie zajmuje się biocybernetyka?

— Ojciec cybernetyki, Norbert Winner w swojej pierwszej książce napisał, że cybernetyka jest nauką o sterowaniu w zwierzętach i maszynach. W momencie gdy wyodrębniła się cybernetyka techniczna, społeczna, ekonomiczna zaistniała potrzeba, by ktoś wrócił do korzeni, do zagadnień inżynierskiego spojrzenia na obiekty żywe i biologię.

Nie wszyscy jednak traktowali tę dziedzinę poważnie. Sam musiałem zapracowywać sobie działaniem na innym gruncie na to, by pozwolono mi zajmować się biocybernetyką. Przez bardzo długi czas moje zainteresowania miały charakter hobbystyczny.

— Czego możemy się spodziewac po tej dziedzinie nauki?

— W wielu przypadkach zdarza się, że problemy, które chcielibyśmy rozwiązywać na gruncie techniki nie znajdują w samej technice dostatecznej bazy. Systemy biologiczne rozwiązują te zadania w sposób optymalny. Poznanie jak pewne funkcje realizuje mózg człowieka, jak sterowane są skrzydła muchy, jak regulowany jest proces przemiany materii u nisko zorganizowanych zwierząt, może nam dostarczyć, jeśli nie gotowych wzorów to źródeł inspiracji. W przyszłości będziemy być może budować rozwiązania doskonalsze niż rozwiązania żywej natury. Dzisiaj jednak obserwowane w przyrodzie systemy stanowią dla nas wzór niedościgłej doskonałości.

— Uważa Pan więc, że jesteśmy w stanie prześcignąć przyrodę?

— Oczywiście! Mamy przecież do dyspozycji lepsze tworzywo. My ludzie, systemy biologiczne, organizmy, jesteśmy zbudowani z tego, co przyroda miała pod ręką, a jest to budulec, który w rękach współczesnego inżyniera wzbudziłby jedynie rozbawienie. Biorąc pod uwagę właściwości mechaniczne, elektryczne, informatyczne elementów naszego ciała to czym dysponuje inżynier jest nieporównanie lepsze. Szybkość przebiegu impulsu nerwowego we włóknie nerwowym nie przekracza 100 m/s a w praktyce wynosi kilka cm/s. Jeśli to porównamy z szybkością sygnału elektrycznego. Szybkość z jaką może pracować komórka nerwowa wchodząca w skład naszego mózgu nie przekracza jednego kHz to znaczy 1000 impulsów w ciągu sekundy. Elementy naszych mikrokomputerów pracują z częstotliwościami znacznie większymi, nawet o kilka rzędów wielkości. Wreszcie problem niezawodności — w naszym mózgu codziennie obumiera kilkanaście do kilkudziesięciu tysięcy komórek, giną bezpowrotnie. W naszych komputerach awarie zdarzają się znacznie rzadziej.

Z drugiej strony fakt, że dysponując tak kiepskim i prymitywnym budulcem przyroda stworzyła tak wspaniałe dzieła, stanowi dla nas powód do podziwu i dopiero do prób naśladowania. Weźmy choćby tylko kwestię niezawodności. Wiemy, że im bardziej złożony system, tym częściej się psuje. Pod czaszką nosimy nieprawdopodobnie skomplikowany system, w skład którego wchodzi kilkanaście miliardów komórek nerwowych. Każda z nich osobno jest elementem zawodnym a jako system tworzą niezawodny mózg. W szczególnie odpowiedzialnych systemach technicznych montuje się nie jeden a dwa, trzy lub więcej komputerów pracujących równolegle aby zabezpieczyć się na wypadek awarii któregoś z nich. Przyroda zrobiła to inaczej, a odpowiedź na pytanie „jak?" jest jednym z najbardziej fascynujących problemów biologicznych i technicznych.

Obcując z wielkimi tworami przyrody takimi jak mózg czy system wzrokowy uczymy się pokory. My technicy lubimy popadać w samozadowolenie, a okazuje się, że to co zrobiła przyroda głupich parę miliardów lat temu jest naprawdę wspaniałe.

— Miała ne to trochę czasu...

— Tak, ale my dopiero teraz, próbując naśladować te systemy w technice, możemy się przekonać jak bardzo są one złożone.

— Patrząc na konstrukcję skrzydeł ptaka czy szkielet człowieka podziwiamy perfekcyjną konstrukcję mechaniczną. Często nie zauważamy natomiast sposobu organizacji informacji w organizmach żywych.

— Już na poziomie pojedynczej komórki toczą się niesłychanie złożone procesy regulacyjne, cybernetyczne, informacyjne. Procesy oddychania wewnętrznego, przemiany materii, sterowania metabolizmem dają się opisać w kategoriach cybernetyki, tak samo jak procesy przetwarzania informacji w procesorach. Elementami uczestniczącymi w tym procesie są pojedyncze molekuły, nośnikiem informacji jest białko strukturalne będące na granicy możliwej do osiągnięcia miniaturyzacji — na poziomie molekularnym.

W medycynie patrzy się zwykle na człowieka przez pryzmat XIX-wiecznej biochemii. Nasz lek to najczęściej pigułka, metody pozyskiwania informacji to głównie analityka chemiczna. Organizm traktuje się jako obiekt, w którym przebiegają procesy chemiczne, w przyszłości będziemy musieli zwrócić uwagę na przebiegające w nim procesy informatyczne. Oczywiście nie jest to ostatnie słowo. Kiedyś Kartezjusz traktował człowieka jako układ mechaniczny, gdy nauka odkryje nowe sfery, okaże się, że i one są obecne w mechanizmie człowieka.

— Programy komputerowe, układy automatyki i sterowania są systemami zdeterminowanymi. Znaczy to, że można przewidzieć jakie parametry przyjmie układ w danej sytuacji. Czy zjawiska przyrodnicze są również zdeterminowane?

— Proces modelowania może również przewidywać zjawiska niezdeterminowane, np. oddziaływania ze strony środowiska. Inna sprawa, czy te zjawiska naprawdę nie dają się przewidzieć, czy my tego nie potrafimy. Jest to jeden z problemów filozofii.

— Czy cybernetyka służy technice czy biologii?

— Cybernetyka oddaje usługi jednej i drugiej stronie. Dzisiaj lepiej wychodzi na tym medycyna, która posiada wiele potrzeb w stosunku do współczesnej techniki, np. w zakresie diagnostyki, terapii, budowy sztucznych narządów. Jeśli cybernetyka jest mostem pomiędzy nauką i techniką, to znacznie częściej maszerują po nim wyroby techniki do biologii i medycyny niż w drugą stronę. Pora jednak na to, by i technika zaczęła częściej korzystać z tego mostu.

— Powstają już przecież pierwsze „fabryki bez ludzi" i w pełni zautomatyzowane linie technologiczne.

— Obecne systemy sterowania i automatyki są budowane w oparciu o doświadczenia inżynierskie. Biocybernetyka nie może sobie rościć pretensji do współautorstwa tych rozwiązań. Dalszy rozwój tych dziedzin techniki może być jednak znacznie łatwiejszy, jeśli skorzysta się ze wzorów jakie daje nam natura. Oto przykład:

W systemach analizy obrazu mamy do czynienia z ilością informacji rzędu kilkudziesięciu milionów bitów na sekundę. Zatka to pamięć nawet dużego komputera. Taka sama ilość informacji rejestrowana przez siatkówkę oka podlega natychmiast — jeszcze w samej siatkówce — ogromnej redukcji. Okazuje się, że z tych wielu milionów bitów informacji do strefy decyzyjnej dociera co najwyżej kilkanaście bitów na sekundę. Obróbka informacji dzieje się poza poziomem naszej świadomości. Mamy prawo twierdzić, że działa tu pewien automatyzm, który można odtworzyć w systemach sterowania.

— Cybernetyka była do niedawna świetną zabawą. Psy, żółwie, wiewiórki cybernetyczne — czy były to tylko zabawki dużych dzieci?

— Myślę, że nie była to tylko zabawa, a mam do tego osobiste powody. Moją pracą magisterską na AGH było zbudowanie psa cybernetycznego. Było to bardzo sympatyczne zwierzątko posiadające między innymi słuch. Inaczej zachowywał się, gdy mówiono do niego „ładny piesek, ładny", a inaczej, gdy wołano „kundel". Fascynacja cybernetycznymi zwierzakami została mi do dzisiaj, jednak obecnie istniejące narzędzia pozwalają tworzyć takie modele w bardziej wdzięcznej substancji. Ta sama zabawa jest możliwa we wnętrzu komputera. Jest przy tym szybsza, łatwiejsza w modyfikacjach i w sumie tańsza. A że odbywa się w nieco ab-

strakcyjnym świecie komputerowego ekranu to chyba sprawa drugorzędna.

Warto byłoby natomiast zaapelować do użytkowników komputerów domowych, aby przerwali kontredans gier komputerowych i spróbowali zabawić się w tworzenie takich stworków cybernetycznych działających na komputerach. Jest wspaniała książka nieżyjącego już profesora Mazura — „Cybernetyczna teoria systemów samodzielnych". Można w niej znaleźć opis prostego modelu czegoś, co działając autonomicznie przypomina zwierzę, człowieka czy może nawet społeczeństwo. Zamodelowałem kiedyś takie „społeczeństwo" na komputerze ODRA 1300. W zależności od warunków stworki walczyły ze sobą lub współdziałały. Daje to niezwykle ciekawe obserwacje, no i można być kreatorem nowych bytów. Każdy posiadacz mikrokomputera ma to w ręku, byle tylko chciał.

— Na następnych stronach „Bajtka" prezentujemy Pański program „Modelowanie rodziny pszczelej". Czy został stworzony dla pszczelarzy, inżynierów, czy może socjologów.

— Na początku była zwykła ciekawość. Czy ten złożony proces da się zamodelować? W efekcie powstał program — znacznie bardziej złożony niż ten publikowany — napisany na „dużym" komputerze. Był on przedmiotem dziesięcioletnich badań.

Model ma bardzo praktyczne ukierunkowanie. Pszczelarz może z jego pomocą przewidywać skutki swoich działań a tym samym może przewidywać optymalne strategie sterowania uwzględniając rozmaite warunki środowiskowe — przede wszystkim pogodę — w sposób bezpieczny, nie zagrażający życiu pszczół. Model może się okazać szczególnie przydatny w sytuacji, kiedy cele, dla których będziemy sterowali rodziną pszczelą będą celami nietradycyjnymi. Od tysiącleci ludzie hodowali pszczoły dla miodu. Dzisiaj przed pszczelarstwem mogą stać zupełnie inne zadania.

Chemizacja rolnictwa spowodowała, że wyginęły dziko żyjące owady zapylające rośliny owadopylne. Jeśli chcemy dzisiaj uzyskać prawidłowe plonowanie sadów, gryki czy łubinu to może się okazać, że jest celowym hodowanie pszczół, do których nawet by się dopłacało a które pełniłyby funkcję usługową w stosunku do rolnictwa. Tradycyjne metody sterowania rodziną pszczelą okazują się w takim przypadku niewystarczające. Nie potrafimy uzyskać maksymalnej liczby pszczół lotnych w stosunkowo krótkim okresie kwitnienia sadów a jednocześnie, żeby przedtem i potem było ich mało. Podobne problemy spotyka się w hodowlach prowadzonych w celu uzyskiwania mleczka, czy stosowanego w farmakologii jadu pszczelego.

Zgodność modelu z rzeczywistymi zjawiskami zachodzącymi w życiu jest bardzo duża. Największy zanotowany błąd wynosił 14%. Oczywiście w przypadku prognozowania dochodzi jeszcze czynnik zmiennych warunków atmosferycznych, a więc błąd może być większy.

— Czy możemy się spodziewać stworzenia tworu o organizacji zbliżonej do organizmu żywego, może samego człowieka?

— Jeśli jeszcze istnieją ludzie, którzy żywią nadzieję, że stworzony zostanie sztuczny człowiek, to stanowią oni mniejszość z co najmniej z dwóch powodów. Po pierwsze dzisiaj znacznie lepiej niż jeszcze kilka lat temu zdajemy sobie sprawę z tego, jak bardzo złożonym systemem jest każdy żywy organizm. Naiwnością i nieuzasadnionym optymizmem byłoby twierdzenie, że potrafimy coś takiego zrobić. Z drugiej strony nie widać rozsądnego celu, dla którego taki duplikat systemu biologicznego miałaby powstać. Z punktu widzenia użyteczności pokraczne roboty przemysłowe, które są karykaturą wyłącznie ręki człowieka, są znacznie bardziej przydatne niż robot podobny do widywanych na filmach fantastyczno-naukowych. Z punktu widzenia inżyniera całość organizmu żywego jest tworem nadmiarowym. Jest w nim mnóstwo rzeczy, które są nam do niczego nieprzydatne. Zbudowanie robota, który wpadałby w złość — a jest to cechą nawet najbardziej prymitywnego zwierzęcia — lub mógłby zakochać się w innym robocie jest po prostu niecelowe.

rozmawiał:
Roman Poznański



UL

MIKROKOMPUTEROWY MODEL RODZINY PSZCZELEJ

Utarł się pogląd, że mikrokomputery mogą być wykorzystywane głównie do zabawy i nauki, a jeśli powierza im się poważne funkcje, to zwykle dotyczą one codziennych prac domowych lub biurowych. Wyraźnie upośledza to inne zastosowania. Warto zatem podejmować próby przełamania tych stereotypów. Prezentowany program pozwala wykorzystać domowy komputer w... pszczelarstwie.

Model powstał w wyniku prowadzonych od 1975 roku badań podstawowych w Zakładzie Biocybernetyki AGH i Zakładzie Pszczelarstwa Akademii Rolniczej w Krakowie. Prace te były częściowo finansowane przez Polską Akademię Nauk. Oczywiście celem badań nie było napisanie programu w BASICU-u, tylko określenie matematycznych prawidłowości w funkcjonowaniu rodziny pszczelej, a także przebadanie przydatności wiedzy na temat cybernetycznych aspektów funkcjonowania wielkiego systemu, jakim jest rodzina pszczela.

Tekst programu napisany jest w języku Simons Basic przeznaczonym dla mikrokomputerów firmy Commodore serii C64 lub C128. Nic jednak nie stoi na przeszkodzie, by po drobnych modyfikacjach opracować wersję podanego programu dla innego, posiadanego przez Czytelnika mikrokomputera. Przeróbki programu dotyczyć w takim przypadku będą głównie instrukcji przeznaczonej do graficznej prezentacji wyników modelowania.

Po uruchomieniu programu komputer zadaje użytkownikowi serię pytań a on musi na nie odpowiedzieć z klawiatury. Pytania te dotyczą rasy pszczół, długowieczności robotnic, horyzontu modelowania, aktualnej struktury rodziny oraz stanu zapasów. Ostatnie pytanie: SKĄD CZYTAĆ DANE O POGODZIE I POŻYTKACH? wymaga wprowadzenia z klawiatury NAZWY wcześniej zapisanego na dysku (lub taśmie) zbioru właśnie warunków atmosferycznych i pożytkach dla jakich użytkownik chce modelować. Zbiór należy nazwać i zapisać na dysku (lub taśmie) przy pomocy programu pomocniczego „ZAPIS-DANYCH".

Po uruchomieniu (komenda RUN) użytkownik wprowadza kolejne dane.

Mając zgromadzone dane można podejmować próby modelowania. Czas trwania modelowania (zwykle dość długi, około kilkunastu minut) zależny jest od założonej liczby kroków, jaką poda się w odpowiedzi na pytanie komputera. Krok to 24 godziny życia realnej rodziny. W trakcie modelowania wyświetlany jest numer aktualnie wykonywanego kroku. Po zakończeniu obliczeń można oglądać wyniki modelowania w postaci wykresów, obrazujących, jak zmieniała się podczas modelowania liczba pszczół, ilość zgromadzonego miodu, powierzchnia dostępnych plastrów itp. Kolejne wykresy wywołuje się naciskając dowolny klawisz.

Ci, którzy próbowali komputerowej symulacji rodziny pszczelej przekonali się, że jest to wprawdzie rozrywka zupełnie innego typu, niż większość znanych gier, jednak z całą pewnością może to być fascynująca przygoda intelektualna — tym bardziej atrakcyjna, jeśli ma się możliwość konfrontowania komputerowych wyników z obserwacją pszczół.

Andrzej Migacz
Ryszard Tadeusiewicz
Zakład Biocybernetyki AGH w Krakowie

PROGRAM 1

```
100 ZZ$=CHR$(147):VV$=CHR$(18):FF$=CHR$
    (146):PRINTZZ$
104 INPUT "KS-ILOSC KROKOW";KS:RZ=1000
108 INPUT "RASA MAT PSZCZ CAU/CAR";RA$
112 IF RA$="CAU" THEN C1=28.:C2=.3:GOTO
    120
116 C1=28.:C2=.3
120 INPUT "GR.WYDAJN.DOB.MAT.PSZCZ. W [
    TYS]";C3
124 INPUT "GR.ROCZ.WYDAJN.MAT.PSZCZ. W
    [TYS]";C4
128 IF C3<=.0 OR C4<=.0 THEN PRINTZZ$"Z
    M I E N  M A T K E ":GOTO 120
132 INPUT "POCZ.WYD.MAT.PSZCZ.W [TYS/DO
    BE]";DY
136 IF DY<=.0 THEN PRINTZZ$"N I E C H
    Z A C Z N I E  C Z E R W I C":GOTO 132
140 INPUT "L.CZERW.WYFR.OD POCZ.SEZ W [
    TYS]";SS
144 INPUT "TM-CZAS ZYCIA PSZCZ.W [DOB.]
    ";TM
148 IF TM<21 THEN PRINT "P S Z C Z O L
    Y  C H O R E ":STOP
152 CZ=TM-21
156 DIM A$(18),B(18),X(18,KS),OM(CZ)
160 A$(1)="DY-PROD.CZERWIU W [TYS/DOBE]
    "
164 A$(2)="Y -LICZBA JAJ W [TYS]"
168 A$(3)="Y1-LICZBA LARW W [TYS]"
172 A$(4)="Y3-LICZBA PSZCZOL W [TYS]"
176 A$(5)="Y4-PSZCZOL 10 DNIOWYCH W [TY
    S]"
180 A$(6)="Y5-PSZCZOL 20 DNIOWYCH W [TY
    S]"
184 A$(7)="Y8-ZBIERACZKI NEKTARU W [TYS
    ]"
```

```
188 A$(8)="YV-ZBIERACZKI PYLKU W [TYS]"
192 A$(9)="W4-NEKTAR LUB SPADZ W [KG/DOBE]"
196 A$(10)="PK-WSP. WSPOLCZ WAGI W [J.W.]"
200 A$(11)="B -WYDRZ.PODZ.PULI W [J.W.]"
204 A$(12)="U0-ZAPAS MIODU NIEZASKLEP. W [KG]"
208 A$(13)="U1-ZAPAS MIODU ZASKLEP.W [KG]"
212 A$(14)="U2-ZAPAS PIERZGI W [KG]"
216 A$(15)="SR-POW.WEZY W [DCM.KWADR.]"
220 A$(16)="X1-SPOZ.MIODU W [KG/DOBE]"
224 A$(17)="PT-TEMP.WSP.WAGI W [J.W.]"
228 A$(18)="X3-NADM.SPOZ.MIODU [KG/DOB]"
232 FOR I=1 TO 18:READ Z:B(I)=Z:NEXT
236 DATA5,5,5.5.2,2,5,5.8,8,4,9,9,9,4,6.6,6
240 PRINTA$(12)::INPUTU0:PRINTA$(13)::INPUTU1
244 PRINTA$(14)::INPUTU2:PRINTA$(2)::INPUTY:PRINTA$(3)::INPUTY1
248 INPUT "Y2-CZERW ZASKLEPIONY W [TYS]";Y2
252 PRINTA$(4);:INPUTY3:PRINTA$(5)::INPUTY4:PRINTA$16);:INPUTY5
256 INPUT "SKAD CZYTAC DANE O POGODZIE I POZYTKACH";N$:OPEN 2,8,2,N$+",S,R"
260 INPUT#2,KP:PRINT "ZAKRES DANYCH SRODOWISKOWYCH";KP
264 DIMTT(KP),TD(KP),Q1(KP/10+1),Q2(KP/10+1),QT(KP/10+1)
268 FOR I=1 TO KP:INPUT#2,TT(I),TD(I):NEXT
272 FOR I=1 TO KP/10+1:INPUT#2,Q1(I),Q2(I),QT(I):NEXT:CLOSE 2
276 V0=0:V1=0:V2=0:SB=0:K=1
280 PRINT AT(15,23)K:GOSUB 344
284 O1=Q1(INT(K/10)+1):O2=Q2(INT(K/10)+1):L=QT(INT(K/10)+1)
288 TA=TT(K):DE=TD(K)
292 IF TA <=10 THEN PT=0
296 IF TA>10 AND TA<=14 THEN PT=.25*(TA-10)
300 IF TA>14 THEN PT=1
304 IF DE=0 THEN PD=1
308 IF DE>0 AND DE<=10 THEN PD=1-.1*DE
312 IF DE>10.THEN PD=0
316 P=PD*PT
320 GOSUB 812:GOSUB 476:GOSUB 500:GOSUB 520:GOSUB 532:GOSUB 592:GOSUB 560
324 X(1,K)=DY:X(2,K)=Y:X(3,K)=Y1:X(4,K)=Y3:X(5,K)=Y4:X(6,K)=Y5
328 X(7,K)=Y0:X(8,K)=YV:X(9,K)=U4:X(10,K)=PK:X(11,K)=B:X(12,K)=U0
332 X(13,K)=U1:X(14,K)=U2:X(15,K)=SR:X(16,K)=X1:X(17,K)=PT:X(18,K)=X3
336 K=K+1:IF K<=KS THEN 280
340 GOTO 824
344 IF K>1 THEN 376
348 FOR I=1 TO 3:OW(I)=Y/3:NEXT
352 FOR I=4 TO 9:OW(I)=Y1/6:NEXT
356 FOR I=10 TO 21:OW(I)=Y2/12:NEXT
360 FOR I=22 TO 31:OW(I)=Y4/10:NEXT
364 FOR I=32 TO 41:OW(I)=Y5/10:NEXT
368 FOR I=42 TO CZ:OW(I)=(Y3-Y4-Y5)/(CZ-41):NEXT
372 GOTO 452
376 J=CZ-1
380 OW(J+1)=OW(J)
384 J=J-1
388 IF J>0 THEN 380
392 PK=.0
396 IF K>11 THEN 408
400 FOR I=1 TO K-1:PK=PK+X(17,I):NEXT:PK=PK/(K-1)
404 GOTO 412
408 FOR I=K-10 TO K-1:PK=PK+X(17,I):NEXT:PK=PK/10.
412 IF X(10,K-1)<=.1 AND X(4,K-1)<=.0 THEN OW(CZ-1)=OW(CZ-1)+OW(CZ)
416 GOSUB 740:OW(1)=DY
420 Y=0:FOR I=1 TO 3:Y=Y+OW(I):NEXT:Y=INT(Y*RZ)/RZ
424 Y1=0:FOR I=4 TO 9:Y1=Y1+OW(I):NEXT:Y1=INT(Y1*RZ)/RZ
428 Y2=0:FOR I=10 TO 21:Y2=Y2+OW(I):NEXT:Y2=INT(Y2*RZ)/RZ
432 Y3=0:FOR I=22 TO CZ:Y3=Y3+OW(I):NEXT:Y3=INT(Y3*RZ)/RZ
436 IF Y3=>1. THEN 448
440 GOSUB 1036:PRINT AT(1,21)VV$;"DZIEN";K;"DAJ PSZCZOLY DO ULA";FF$
444 KS=K
448 Y4=0:FOR I=22 TO 31:Y4=Y4+OW(I):NEXT:Y4=INT(Y4*RZ)/RZ
452 Y5=0:FOR I=32 TO 41:Y5=Y5+OW(I):NEXT:Y5=INT(Y5*RZ)/RZ
456 YP=Y3-Y4-Y5
460 IF YP>.6*Y3 THEN YP=.6*Y3
464 YV=.2*YP
468 Y0=.8*YP
472 RETURN
476 X1=4.2E-3*Y3:X2=20.E-3*Y1:X3=.0
480 IF (Y+Y1) >.0 AND PK>0 THEN X3=10.E-3*Y4
484 X4=0
488 IF K=1 THEN RETURN
492 IF X(15,K-1)>0 THEN X4=1.E-2*Y5*P
496 RETURN
500 X0=1.E-3*L*P*(Y0+YV):U4=12.E-3*L*P*Y0:GU=.75*O1-X0
504 IF U4>GU THEN U4=GU
508 U5=10.E-3*L*P*YV
512 IF U5>O2 THEN U5=O2
516 RETURN
520 U0=U0+U4+V0:U1=U1+V1:U2=U2+U5+V2
524 IF U2<0 THEN U2=0
528 RETURN
532 IF U0=>X1+X2+X3+X4 THEN RETURN
536 IF U0=>X1+X2 THEN 552
540 IF RA$="CAU" THEN X3=.0:RETURN
544 X3=.2*X3:X4=.1*X4
548 RETURN
552 DU=U0-X1-X2:X3=DU*X3/(X3+X4):X4=DU-X3
556 RETURN
560 SY=1.25*(Y+Y1+Y2):SU=3.*(1.5*U0+U1+1.5*U2):SZ=SY+SU
564 SB=SB+60.*U3:SR=32.-SB
568 IF SR<0 THEN SR=0
572 SD=160.-SR
576 IF SZ<SD THEN RETURN
580 SD=SD+16.
584 GOSUB 1036:PRINT AT(1,21)VV$;"DZIEN";K;"DODANO 2 PLASTRY";FF$
588 RETURN
592 X5=.1*X1:IF X5>U2 THEN X5=U2
596 X6=.75*X2:IF X6>U2-X5 THEN X6=U2-X5
600 IF X6<0 THEN X6=0
604 G=U2-X6-X5:IF G<0 THEN G=0
608 X5=.1*X1:IF X5>U2 THEN X5=U2
612 X6=.75*X2:IF X6>U2-X5 THEN X6=U2-X5
616 X8=.75*X4
620 G=U2-X6-X5:IF G<0 THEN G=0
624 X7=.75*X3:IF X7>G THEN X7=G
628 G=G-X7:IF G<0 THEN G=0
632 GOSUB 1036:PRINT AT(1,21)VV$;"DZIEN";K;"DODANO 2 KG SYROPU";FF$
636 P2=U0-X1-X2-X3-X4:IFP2<=0. THEN P2=.0
640 IF (X4<=0) AND (P2<=2) THEN 668
644 U3=.3E-2*Y5
648 IF U3>X(15,K+1)*1.25E-2 THEN U3=1.25E-2*X(15,K-1)
652 IF U3>.25*P2 THEN U3=.25*P2
656 IF U3>20.*P3 THEN U3=20.*P3
660 IF U3>5.E-2*Y5 THEN U3=5.E-2*Y5
664 GOTO 672
668 U3=0
672 XP=U3*4.:XD=5.E-2*U3:U0=U0-X1-X2-X3-X4-XP
676 IF U0<0 THEN 692
680 UA=.1*Y3:IF UA>.1*U0 THEN UA=.1*U0
684 XS=.25*UA:U0=U0-1.25*UA:U1=U1+UA
688 GOTO 716
692 D=U0:U0=0:U1=U1+D:IFU1=>0 THEN 716
696 U1=0:U0=D+2
700 GOSUB 1036:PRINT AT(1,21)VV$;"DZIEN";K;"DODANO 2 KG SYROPU";FF$
704 IF U0>0 THEN 716
708 GOSUB 1036:PRINT AT(1,21)VV$;"TO JEDNAK MALO";FF$
712 KS=K
716 U2=U2-X5-X6-X7-X8-XD:IF U2<0 THEN U2=0
720 IF U1<10 OR U4<.5 THEN RETURN
724 UZ=U1-5.
728 GOSUB 1036:PRINT AT(1,21)VV$:"DZIEN";K;"ODEBRANO 5KG MIODU";FF$
732 U1=5.
736 RETURN
740 IF K=1 THEN RETURN
744 SA=0.
748 IF K>11 THEN 760
752 FOR I=1 TO K-1:SA=SA+X(18,I):NEXT:SA=SA/10
756 GOTO 764
760 FOR I=K-10 TO K-1:SA=SA+X(18,I):NEXT:SA=SA/10
764 IF X(10,K-1)<=.0 OR X(14,K-1)<=0 THEN 796
768 D1=C1*SA+C2
772 IF DY<.0 THEN DY=.0
776 IF DY>C3 THEN DY=C3
780 IF X(1,K-1)=.0 THEN 800
784 IF DY<=.95*X(1,K-1) THEN DY=.95*X(1,K-1)
788 IF DY=>2.0*X(1,K-1) THEN DY=2.0*X(1,K-1)
792 GOTO 800
796 DY=.95*X(1,K-1)*PK
800 IF DY<=1.E-2 THEN DY=.0
804 DY=INT(DY*RZ)/RZ
808 RETURN
812 SS=SS+DY:B=1-(1/C4)*SS
816 IF B<=.0 THEN B=.0
820 RETURN
824 POKE 53280,2:GG=0
828 INPUT "CZY USREDNIAC (TAK/NIE)":A$:IF A$="NIE" THEN 844
832 FORI=1TO18:FORJ=1TOKS/10:S=0:FORK=(J-1)*10+1TOJ*10:S=S+X(I,K):NEXTK
836 FORK=(J-1)*10+1 TO J*10:X(I,K)=.1*S:NEXT K,J,I
840 PRINT "USREDNIONO"
844 GET A$: IF A$="" THEN 844
848 GG=GG+1:IF GG=19 THEN GG=1
852 IF GG=9 OR GG=10 THEN 848
856 E=0
860 POKE 53280,B(GG):HIRES1,B(GG)
864 LINE 50,180,290,180,1
868 FOR K=50 TO 290 STEP 40
872 LINE K,180,K,184,1
876 NEXT
880 TEXT 46,185,"0",1,1,8
884 TEXT 82,185,"20",1,1,8
888 TEXT 122,185,"40",1,1,8
892 TEXT 162,185,"60",1,1,8
896 TEXT 202,185,"80",1,1,8
900 TEXT 238,185,"100",1,1,8
904 TEXT 278,185,"120",1,1,8
908 TEXT 287,170,"K",1,1,8
912 LINE 50,180,50,20,1
916 FOR K=180 TO 20 STEP -40
920 LINE 50,K,46,K,1
924 NEXT
928 GOSUB 972
932 GX=50
936 GY=180
940 FOR K=1 TO KS
944 G0=180-ABS(X(GG,K))/(4*XN*10^E)*160
948 G1=2*K+50
952 LINE GX,GY,G1,G0,1
956 GX=G1
960 GY=G0
964 NEXT
968 GOTO 844
972 XM=0
976 FOR K=1 TO KS
980 IF X(GG,K)>XM THEN XM=X(GG,K)
984 NEXT
988 IF XM<4 THEN 1028
992 IF XM>40 THEN 1032
996 XN=INT(XM/4)+1
1000 FOR K=0 TO 4
1004 X$=STR$(K*XN)+"E"+STR$(E)
1008 TEXT 0,176-K*40,X$,1,1,8
1012 NEXT
1016 TEXT 55,16,LEFT$(A$(GG),2),1,1,8
1020 TEXT 5,5,A$(GG),1,1,8
1024 RETURN
1028 XM=XM*10:E=E-1:GOTO988
1032 XM=XM/10:E=E+1:GOTO992
1036 PRINT AT(1,21)"                    ":RETURN
```

PROGRAM 2

```
100 REM *** ZAPIS  DANYCH ***
105 :
110 INPUT "DLA ILU DNI ZADAJESZ DANE";KP
115 DIM TT(KP),TD(KP),Q1(KP/10+1),Q2(KP/10+1),QT(KP/10+1)
120 INPUT "JAK SIE MA NAZYWAC ZBIOR Z DANYMI";N$
125 OPEN 2,8,2,N$+",S,W": OPEN 15,8,15
130 FOR I=1 TO KP
135 PRINT "DZIEN":I;"TEMPERATURA I OPADY"
140 INPUT TT(I),TD(I)
145 NEXT
150 PRINT "TERAZ DANE FENOLOGICZNE W DEKADACH"
155 FOR I=1 TO KP/10+1
160 PRINT "DEKADA":I;" Q1,Q2,QT"
165 INPUT Q1(I),Q2(I),QT(I)
170 NEXT
175 INPUT"CZY WSZYSTKO W PORZADKU":A$
180 PRINT#2,KP
185 IF A$="NIE" THEN 130
190 FOR I=1 TO KP
195 PRINT#2,TT(I);Z$:TD(I)
200 GOSUB 310
205 NEXT
210 FOR I=1 TO KP/10+1
215 PRINT#2,Q1(I):Z$:Q2(I);Z$:QT(I)
220 GOSUB 310
225 NEXT
230 CLOSE 2:CLOSE 15
235 INPUT "CZY CHCESZ PROBNY WYDRUK DANYCH":A$
240 IF A$="NIE" THEN STOP
245 OPEN 4,4 : CMD 4
250 PRINT "ZAWARTOSC ZBIORU ":N$:" POGODA"
255 FOR I=1 TO KP
260 PRINT I,TT(I),TD(I)
265 NEXT
270 IF I=1 THEN PRINT Q1(I),Q2(I),QT(I):
275 PRINT "ZAWARTOSC ZBIORU ":N$:" FENOLOGIA"
280 FOR I=1 TO KP/10+1
285 PRINT Q1(I),Q2(I),QT(I)
290 NEXT
295 PRINT#4
300 CLOSE 4
305 END
310 INPUT#15,EN,EM$,ET,ES
315 IF EN>21 THEN PRINT "BLAD ZAPISU NA DYSKU":EN,EM$,ET,ES:STOP
320 RETURN
READY
```

ZAMIAST PIRACKIEJ FLAGI

Seria artykułów odkrywających między innymi tajniki zabezpieczania programów na ZX Spectrum, których druk rozpoczynamy w tym numerze, stała się w naszej redakcji przyczyną gorących dyskusji na temat włamywania się do programów (i oczywiście bezprawnego ich kopiowania). Uważamy, że kilka różnych poglądów na ten temat zasługuje na uwagę.

Nie byłbym sobą, gdybym nie wtrącił kilku zdań, wyjaśniających istotę sprawy dla osób, które jeszcze nie miały okazji zetknąć się z problemem. Otóż napisanie dużego programu (np. kompilatora lub dobrej gry) wymaga ogromnego nakładu pracy (mogą to być miesiące lub nawet lata pracy). Gotowy produkt może być powielany (kopiowany) i każda z kopii ma taką samą wartość użytkową jak oryginał. Pozwala to autorowi sprzedać program wielu użytkownikom i uzyskać godziwą rekompensatę za włożony trud. Jednak nieuczciwy nabywca może także robić kopie i sprzedawać je na własną rękę, osiągając korzyści, które mu się nie należą (oczywiście osiąga je kosztem autora). Aby temu zapobiec, wielu autorów i firm produkujących oprogramowanie wstawia do swoich programów fragmenty mające uniemożliwić zrobienie poprawnej kopii. Od czego jednak pomysłowość. Trzeba taki fragment zlokalizować i zneutralizować (czyli złamać protekcję) i dalej hulaj dusza, piekła nie ma. Żeby jednak robić takie sztuki w miarę sprawnie, potrzebny jest pewien zasób wiedzy. A między innymi taką właśnie wiedzę można znaleźć w artykule „Od środka", więc wracamy do redakcji „Bajtka".

Bardzo szybko uzgodniliśmy, że nie można odrzucać dobrego, fachowego artykułu tylko dlatego, że wiedza w nim zawarta może ułatwić złamanie zabezpieczenia programu i w konsekwencji jego skopiowanie wbrew woli autora. Nie można przecież mieć pretensji do producenta zapałek o to, że istnieją podpalacze. Gdyby zrezygnować z produkcji zapałek, to ucierpią na tym przede wszystkim zwykli uczciwi ludzie, używający ich do zaspakajania swoich codziennych potrzeb. A Czytelnicy „Bajtka", to przecież właśnie zwykli uczciwi ludzie.

Jednak na tym dyskusja się nie zakończyła, gdyż jak się dalej okaże temat jest bardzo gorący. Bezpośrednim pretekstem stał się pomysł przywrócenia do łask pirackiej flagi z kilku poprzednich numerów, oraz propozycja tytułu „Coś dla włamywaczy". Argumenty płynęły dwutorowo: po pierwsze, włamywanie się do programów staje się w świecie powszechną zabawą, stymulującą intelektualnie próbą sił między dwoma programistami, pośrednio mobilizującą do pogłębiania wiedzy i zdobywania nowych umiejętności. Włączmy się do tej zabawy i nazywajmy ją po imieniu. Precz z obłudą, piracka flaga na maszt! Drugi nurt był praktyczny: bardzo wielu tzw. „sprzedawców" oprogramowania, czy też „sklepów" informatycznych, to zwykli złodzieje, którzy złamali zabezpieczenia w cudzych programach i teraz handlują nielegalnymi kopiami. Po co dawać chleb takiemu złodziejowi-piratowi, niech przeciętny zjadacz chleba nauczy się tego samego, co oni i robi sobie kopie sam, za darmo. Jednym słowem, precz z piratami-dorobkiewiczami, amatorzy flaga na maszt.

Przyznam, że zaliczam się do tych, których szokuje fakt wydawania przez nasze władze zezwoleń na działalność nie mającą nawet pozorów uczciwości, i perspektywa rychłego rozprawienia się ze zorganizowanym złodziejstwem jest tu nęcąca, ale czy można robić to za wszelką cenę? Przypatrzmy się proponowanemu schematowi działania: cel — doprowadzić do przestrzegania pewnych zasad: metoda osiągnięcia celu — złamanie tychże zasad. Widać wyraźnie, że przeszliśmy już do kontrargumentów, pierwszy właśnie przeczytaliście. Drugi jest chyba dużo bardziej zasadniczy: w naszym kraju, który właśnie zaczyna zbliżać się do upowszechnienia informatyki (przynajmniej jej części reprezentowanej przez mikrokomputery), praktycznie nie funkcjonuje żadne, ani formalne, ani zwyczajowe prawo własności oprogramowania. Komputer, owszem tak, za komputer trzeba zapłacić, i to dużo. Ale program, po co kupować, zawsze się od kogoś odegra. Tego typu podejście skutecznie blokuje powstawanie czegoś, co na świecie nazywa się przemysłem produkującym oprogramowanie — no bo kto zechce robić coś, co społecznie nie jest uważane za wartościowe, coś, co jest w randze prawie śmiecia, przekazywane z ręki do ręki praktycznie za darmo. Być może piractwo jest świetną, niezbyt szkodliwą zabawą, ale w społeczeństwach, w których ktoś, kto ukradł program staje przed sądem, gdzie powszechnie dostrzega się wartość programu, gdzie są nawyki szanowania tej wartości. U nas nawyki w tej dziedzinie są fatalne, i dlatego ostatecznie uznaliśmy, że powtarzanie (w sumie dość niewinnego) żartu z piracką dyskietką na maszcie może być odczytane wbrew intencjom Autorów, jako sankcjonowanie, czy nawet popieranie grabieży programów. Dlatego zamiast flagi na maszcie dostaliśmy te kilka uwag. Mam nadzieję, że udało mi się przekonać Was, że kopiując cudzy program nie przyspieszacie rozwoju polskiej informatyki i jej wyjścia z zacofania lecz robicie właśnie coś przeciwnego. Ja wiem, że w tej chwili jest bardzo źle, nie mam też zamiaru udawać jedynego sprawiedliwego, bo nim nie jestem. Chcielibyśmy jednak, i ja i cała redakcja „Bajtka", aby do świadomości każdego Polaka dotarło, że za programy trzeba płacić. Oczywiście tylko i wyłącznie ich autorom.

Andrzej Pilaszek

OD ŚRODKA CZ. 1

Mało kto lubi programy, które przy pierwszym lepszym błędzie lub wciśnięciu klawisza BREAK czyszczą całą pamięć komputera, nie pozostawiając po sobie żadnego śladu, albo „zawieszają się", zmuszając do wciśnięcia RESET. Sytuacja przestaje być zabawna, gdy mamy jakiś dobry program użytkowy, który chcemy przystosować do nietypowego sprzętu (lub grę do rzadko spotykanego joysticka) lub gdy chcemy zmienić w programie wszystkie teksty angielskie na polskie, a program nie daje się zatrzymać.

Chcielibyśmy przedstawić wam kilkuodcinkowy cykl artykułów. Kolejno więc przedstawimy niezbędne przy takiej pracy informacje o twoim (lub może pożyczonym) komputerze, takie jak mapa pamięci, sposób zapisu w pamięci poszczególnych linii BASIC-a, ważne zmienne systemowe, itp. Później zabierzemy się do wczytywania programów i bloków danych z taśmy w „bezpieczny" sposób, tzn. tak, by się nie uruchomiły i by można było obejrzeć ich zawartość. W końcu zajmiemy się także unieszkodliwianiem zabezpieczeń pisanych w języku wewnętrznym. Postaramy się wszystko to ilustrować konkretnymi przykładami, w oparciu o znane programy. Mamy nadzieję, że nasz wysiłek nie pójdzie na marne i ty także nauczysz się dostawać bez przeszkód do każdego programu.

Zacznijmy więc od podziału pamięci.

Ogólnie, pamięć podzielona jest na dwie główne części: ROM i RAM. ROM zajmuje adresy 0 — 16383, RAM natomiast adresy 16384 — 65535. Zawartością ROM-u nie będziemy się na razie zajmować, lecz za to przyjrzymy się dokładniej pamięci RAM. Jest ona podzielona na bloki spełniające różne funkcje w systemie BASIC-a (rys.1).

Pierwszym z nich jest obszar pamięci ekranu. Począwszy od adresu 16384 znajduje się tzw. „display file", czyli obszar, w którym przechowywane są informacje o tym, czy kolejne punkty ekranu są zapalone, czy zgaszone. Zajmuje to 6144 bajtów. Następne 768 bajtów (od adresu 22528) to komórki pamięci określające kolory kolejnych pól ekranu (8 X 8 punktów). Obszar od adresu 23296 do 23551 to bufor drukarki. Jest on wykorzystywany tylko podczas współpracy komputera z drukarką. Jeśli nie używasz instrukcji dotyczących drukarki (takich jak LLIST, LPRINT, COPY), to

TOS

ZX Spectrum zaprojektowane zostało z myślą o współpracy z magnetofonem lub microdrive'm. Nie oznacza to, że zrezygnowano całkowicie z prac nad stacją dysków. Pamięci dyskowe do ZX Spectrum powstały w wielu firmach i charakteryzują się różnorodnością zastosowanych w nich rozwiązań technicznych. Jednym z najciekawszych urządzeń tego typu jest 3-calowa stacja dysków Timex, umożliwiająca jednoczesne podłączenie do Spectrum 4 napędów dyskowych sterowanych wspólnym kontrolerem. Użytkownik zyskuje w ten sposób szybki i niezawodny dostęp do ponad 500 kB dodatkowej pamięci.

KLAN SPECTRUM

jego zawartość nie ulega zmianie, możesz go więc wykorzystać do innych celów. Pamiętaj jednak, że użycie którejś z tych instrukcji, nawet bez podłączonej drukarki, wprowadza w tym obszarze zmiany.

Następnym fragmentem pamięci RAM są zmienne systemowe. Są to komórki pamięci wykorzystywane przez system do pamiętania niezbędnych do jego prawidłowego działania danych, takich jak np. adresy tzw. ruchomych bloków pamięci (o których zaraz powiemy), informacje o wykonywaniu programu w BASIC-u, tzn. która linia jest wykonywana, do której ma nastąpić skok, czy wystąpiły jakieś błędy, itp. W obszarze tym znajdują się także zmienne (tzn. komórki spełniające te funkcje), zawierające kod ostatnio wciśniętego klawisza, długość „beep" klawiatury i wiele jeszcze innych. Dokładniej zajmiemy się nimi później.

Bezpośrednio za zmiennymi systemowymi, które kończą się pod adresem 23733, zaczynają się tzw. ruchome obszary pamięci. Oznacza to, że adresy ich początków i końców (a także długość) mogą się zmieniać, w zależności od tego, czy są podłączone jakieś urządzenia zewnętrzne, jak długi jest program w BASIC-u, ile tworzy zmiennych, itp. Adresy ruchomych bloków RAM-u znajdują się w odpowiednich zmiennych systemowych.

Na rys. 1 i rys. 2 liczba pod strzałką oznacza adres początku wskazywanego bloku. Jeśli adres ten jest ruchomy, to zamiast liczby zapisana jest nazwa zmiennej systemowej zawierającej ten adres, oraz w nawiasie — adres tej zmiennej. W nawiasie kwadratowym znajduje się wartość tej zmiennej ustalana zaraz po włączeniu komputera (lub wykonaniu RESET), ale bez podłączonych żadnych urządzeń zewnętrznych (czyli jeśli odczytamy wartość zmiennej **PROG**, wykonując

PRINT PEEK 23635 + 256 * PEEK 23636,

to otrzymamy wartość **23755**).

Jeśli do twojego komputera podłączony jest „Interface 1" lub interface innej szybkiej pamięci masowej, to od adresu 23734 do adresu o 1 mniejszego niż zawartość zmiennej CHANS, znajduje się „mapa microdrive'u" — obszar wykorzystywany jako bufor do transmisji danych, jako zbiór dodatkowych zmiennych systemowych itp. Jeśli nie jest podłączone żadne z tych urządzeń, obszar ten po prostu nie istnieje — zmienna CHANS zawiera adres 23734. Określa ona początek bloku pamięci, w którym zawarte są informacje o istniejących kanałach. Są one konieczne do prawidłowego działania instrukcji PRINT, LIST, INPUT i podobnych. W ostatniej komórce tego obszaru znajduje się liczba 128 (heksadecymalnie 80), sygnalizująca koniec tego bloku (jest to tzw. znacznik końca). Następny obszar pamięci zawiera tekst wpisanego programu w BASIC-u. Adres jego początku pamiętany jest w zmiennej PROG. Bezpośrednio za tekstem programu (od adresu wskazywanego przez zmienną VARS), znajduje się obszar, w którym interpreter umieszcza zmienne tworzone przez program. Jest on zakończony znacznikiem końca. Następnie, począwszy od adresu zawartego w zmiennej E_LINE znajduje się obszar wykorzystywany podczas edycji linii BASIC-a oraz wpisywaniu komend z klawiatury (tzn. gdy na dole ekranu miga kursor i wpisujemy instrukcje w BASIC-u). Na końcu tego obszaru znajdują się dwa bajty, o zawartościach: 13 (ENTER — koniec linii) i 128 (koniec tego obszaru). Zaraz potem, od adresu wskazywanego przez zmienną systemową WORKSP, znajduje się podobny obszar, ale służący do wpisywania danych podczas wykonywania przez interpreter instrukcji INPUT (zakończony znakiem ENTER).

Za buforem INPUT (który jest automatycznie kasowany po wykonaniu tej instrukcji) znajduje się „chwilowa przestrzeń pracy" — miejsce pamięci wykorzystywane do najrozmaitszych celów. Tam między innymi ładowane są nagłówki wczytywanych z taśmy programów, tam jest wczytywany program umieszczany w pamięci przez MERGE"", zanim zostanie dołączony do już istniejącego programu. Obszar ten jest wykorzystywany wtedy, gdy na pewien czas potrzebujemy trochę wolnej pamięci, ale tylko do chwilowego wykorzystania — potem nie jest dla nas ważne co się z jej zawartością stanie.

Od adresu wskazywanego przez STKBOT, znajduje się stos kalkulatora. Są tam odkładane liczby w trakcie wykonywania obliczeń przez interpreter BASIC-a. Stos ten rozrasta się w górę pamięci, tzn. w kierunku coraz wyższych adresów. Zmienna systemowa STKEND określa jego koniec. Za nim znajduje się obszar nie wykorzystywanej pamięci.

Do systemu BASIC-a należy jednak obszar aż do komórki pamięci wskazywanej przez zmienną systemową RAMTOP. Pod tym adresem znajduje się liczba 62 (3E hex), która oznacza koniec obszaru wykorzystywanego przez BASIC.

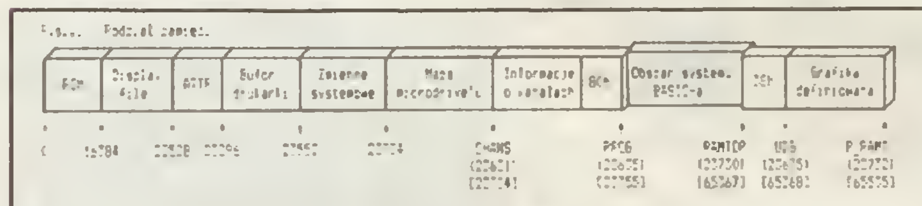
Idąc teraz w dół pamięci, trafiamy na jeden bajt nie wykorzystywany *). Zaraz za tym bajtem (idąc cały czas w dół pamięci), zaczyna się „stos GOSUB". Odkładane są na nim numery linii programu, z których zostały wykonane instrukcje skoku do podprogramu, aby interpreter wiedział dokąd ma „wrócić" instrukcja RETURN. Jeżeli interpreter nie znajduje się w żadnym podprogramie (wywołanym właśnie przez GOSUB), to stos ten po prostu nie istnieje — nie jest na nim zapisana żadna wartość. Niżej znajduje się stos maszynowy, wykorzystywany bezpośrednio przez mikroprocesor. Obydwa te stosy są odkładane w dół pamięci.

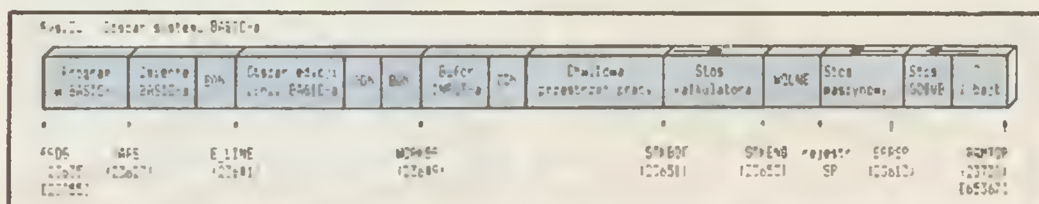
Specjalną rolę pełni zmienna systemowa ERRSP. Procedura obsługująca błąd BASIC-a (wywoływana przez rozkaz mikroprocesora RST 8) umieszcza wartość tej zmiennej w rejestrze SP, po czym wykonuje RET, odczytując w ten sposób ostatni zapisany na stosie adres (podczas wykonywania programu jest on równy 4867). Pod tym adresem w ROM-ie znajduje się procedura drukująca komunikat o błędzie.

Powyżej komórki wskazywanej przez RAMTOP znajduje się 168 bajtów zarezerwowanych na definicje znaków UDG (można je zlikwidować np. przez CLEAR 65535). Adres ostatniej komórki pamięci (równy 65535, jeśli twój komputer jest całkowicie sprawny) jest pamiętany w zmiennej P_RAMT. Jeżeli część pamięci RAM jest uszkodzona, to zmienna ta zawiera adres ostatniej sprawnej komórki.

To by było wszystko, jeśli chodzi o podział pamięci Spectrum. Za miesiąc zajmiemy się już włamywaniem do programów napisanych w BASIC-u oraz nagłówkami zbiorów zapisanych na taśmie.

Tomasz Surmacz
Robert Dudzik





***) Bajt ten tworzy wraz z bajtem wskazywanym przez RAMTOP jakby jedną, dwubajtową liczbę (jest jej młodszym bajtem), konieczną do prawidłowego działania instrukcji RETURN. Gdy podczas jej wykonywania stos GOSUB-ów będzie już pusty, to liczba ta spełni rolę jego przedłużenia. Ponieważ jednak jest ona większa niż 15872 (62*256), a linie BASIC-a nie posiadają tak wysokiej numeracji, więc zostanie to wykryte jako błąd i zasygnalizowane przez komunikat „RETURN without GOSUB".**

Pamięć dyskową Timex dołączamy do złącza krawędziowego ZX Spectrum za pośrednictwem interfejsu. Składa się ona z zasilacza, kontrolera i napędów dyskowych umieszczonych w oddzielnych obudowach. Wewnątrz kontrolera znajduje się niemal osobny komputer: mikroprocesor Z80, 16 kB pamięci, sterownik dysku i układy wejścia-wyjścia. Korzystanie z urządzenia odbywa się pod kontrolą dyskowego systemu operacyjnego TOS (Timex Operating System), który ze względu na bogactwo występujących w nim funkcji można śmiało określić angielskim terminem „user friendly" — sprzyjający użytkownikowi.

Już dołączenie jednego napędu znacznie zwiększa możliwości Spectrum. Na jednej stronie dyskietki znajduje się 40 ścieżek podzielonych na sektory po 256 bajtów, co daje łącznie ok. 160 kB. Sam system operacyjny zajmuje ok. 20 kB (16 kB dla systemu oraz 4 kB katalog dysku), a więc dla użytkownika pozostaje 140 kB. Czysta dyskietka przygotowywana jest do pracy przez procedurę formatowania, podczas której tworzone są ścieżki i sektory oraz zapisywany jest sam system operacyjny. Formatowanie jest możliwe po umieszczeniu w napędzie dyskietki już sformatowanej, czyli zawierającej TOS, np. dyskietki demonstracyjnej. Następną czynnością jest podanie komendy

Format * "nazwa napędu" TO "nazwa dysku"

Nazwami napędów są litery A, B, C, D. W składni komend TOS wykorzystano słowa kluczowe występujące w BASIC-u. Przed formatowaniem system żąda potwierdzenia polecenia wyświetlając na ekranie zapytanie: Format disk in drive A Y/N ? ... dyskietki wzorcowej na czystą. Ostrożność ta jest niezbędna, bowiem procedura ta niszczy cały poprzedni zapis, co grozi utraceniem dość cennych danych lub programów. Po około 30 sekundach dyskietka gotowa jest do pracy.

Wszystkie zbiory zapisywane na dysku zorganizowane są w hierarchiczny system zawierający katalog główny oraz podkatalogi. W katalogu głównym oraz każdym podkatalogu umieszczony jest spis zbiorów lub podkatalogów kolejnego poziomu. Struktura ta przypomina rozgałęzione korzenie drzewa.

Katalog zbiorów wyświetlany instrukcją CAT * zawiera szereg istotnych informacji: nazwa aktualnego katalogu, numer aktualnego poziomu w strukturze, nazwę napędu, nazwę, typ, długość, alokację i atrybuty zbiorów. Dolny wiersz informuje o stanie wykorzystania wolnego miejsca. Katalog wyświetlany jest do końca. Jeśli ilość nazw zbiorów przekracza ilość wierszy ekranu, to następuje scroll. Klawisz S zatrzymuje automatyczny scroll w dowolnym miejscu, a Q uruchamia go ponownie.

Nazwa zbioru składa się z ciągu 8 znaków oraz trzyliterowego, oddzielonego kropką rozszerzenia (opcja), oznaczającego typ, np. wszystkim programom w BASIC-u nadamy rozszerzenie .BAS, a maszynowym .COD. Dla TOS zastrzeżone są rozszerzenia DIR (katalog) oraz SCP (Serial Communication Ports). Wszystkie litery w nazwie zamieniane są automatycznie na duże.

Do zapisania lub odczytania programu z dysku nie wystarcza samo określenie jego nazwy. Większość operacji dostępnych w TOS wymaga podania wszystkich nazw podkatalogów występujących na drodze pomiędzy aktualnym katalogiem, a zbiorem docelowym. Poszczególne nazwy składowe tworzą ścieżkę nazw (łącznie zwaną ścieżką dostępu), którą dalej będziemy oznaczać skrótem ... Na przykład SPECTRUM GRY.DIR BRIDGE.BAS wskazuje sposób dojścia z katalogu głównego do programu BRIDGE.BAS. Inaczej mówiąc ścieżka dostępu jest złożoną nazwą umożliwiającą odszukanie zbioru w hierarchicznej strukturze. Składnia komend dyskowych transmitujących dane lub programy między dyskiem a kompute-

JAK MALOWAĆ? (cz.2)

Idealnym rozwiązaniem byłoby stworzenie algorytmu zamalowywania dowolnego obszaru zamkniętego. Poniżej przedstawiam jeden z nich. Pomysł jest ogólnie znany (Sinclair User) i mimo to, że procedura działa wolno, jest ona bardzo krótka i mało skomplikowana.

Każdemu punktowi ekranu przyporządkujmy jego współrzędne w układzie tak, jak na ekranie. Wypełnianie rozpocznie się od ostatniego postawionego punktu, jego współrzędne odczytane ze zmiennej systemowej CO-ORDS (23677 i 23678). Główna pętla programu to linie 120-180. Pod uwagę bierzemy punkt o współrzędnych (x,y) — zapisujemy x jako xp i y jako yp. Teraz patrzymy na punkt powyżej rozpatrywanego i — gdy jest zgaszony — zapalamy go zapamiętując jego współrzędne w tablicy i zwiększamy wskaźnik p określający, ile mamy zapamiętanych punktów. Jeżeli punkt jest zapalony, patrzymy na punkt poniżej (xp, yp) i ewentualnie zapalamy go (podprogram 200). Postępowanie powtarzamy dla punktu z lewej i z prawej strony. Następnie sprawdzamy, czy p jest równe zero. Jeśli tak, to zapełnianie zostało zakończone (żaden z podstawionych punktów nie ma zgaszonego sąsiada). Jeśli nie, odczytujemy z tablicy współrzędne ostatnio zapalonego punktu (w tym przypadku jest to punkt położony po prawej punktu (xp, yp)) i jego bierzemy pod uwagę przebiegając linie 150-180. Jeśli natrafimy na punkt, którego wszyscy sąsiedzi są zapaleni, cofamy się w tablicy zmniejszając p i szukając punktu, którego któryś z sąsiadów nie jest zapalony.

W ten sposób zamalujemy każdy obszar zamknięty przez postawienie wewnątrz niego punktu i skok do linii 100. Jedynym ograniczeniem jest rozmiar tablicy. Jeżeli ilość zapamiętanych i nie usuniętych punktów przekroczy 2000, program zatrzyma się komunikatem Subscript wrong. Na ekranie mamy 256x176=45056 punktów i zapamiętanie wszystkich byłoby niemożliwe, i chociaż program „cofając się" (zmniejszając p) oszczędnie gospodaruje pamięcią, to jednak nie zamaluje całego ekranu nawet startując ze środka.

Dla przykładu: obrazkiem do zamalowania będzie miś. Narysujemy go, wpisując:

```
15 LET su=0
10 FOR a=I TO I0
20 READ x,y,r
30 CIRCLE x,y,r: LET su=su+x+y+r
40 NEXTa: IF su<>2350 THEN PRINT "Złe dane": STOP
50 DATA 128, 118, 30, 167, 140, 15, 89, 140, 15, 115,
      125, 5, 141, 125, 5, 128, 58, 30, 95, 77, 8, 161,
      77, 8, 161, 39, 8, 95, 39, 8
55 LET su=0
60 FOR a=I TO I0
65 READ x,y
70 PLOT OVER I; x,y: LET su=su+x+y
80 NEXT a: IF su<>2056 THEN PRINT "Złe dane": STOP
90 DATA 103, 134, 154, 134, 128, 88, 102, 72, 155, 72,
      156, 72, 102, 43, 102, 43, 154, 43, 155, 43
95 PLOT 128, 86
```

przed programem zamalowywania (listing 1). Miś składa się z kółek (linie 5—50) z przejściami między nimi (linie 55—90). Start zapełniania podaje linia 95.

Widać, że zamalowywanie trwa bardzo długo. Istotne przyspieszenie uzyskamy pisząc ten sam program w asemblerze (listing 2). Schemat działania jest taki sam, jak poprzednio. Zamiast tablicy użyjemy stosu systemowego. Zamiast znacznika długości (poprzednio p) na początku położymy na stos liczbę 255. Odczytanie jej ze stosu będzie równoznaczne z zakończeniem malowania (poprzednio p=0).

Procedura odwołuje się do ROM-u Spectrum wykorzystując zawarte tam procedury PLOT (8933) i POINT (8910), oraz FP-TO-A (11733). Współrzędne punktów przechowywane są odpowiednio: x w C, y w B, xp w L i yp w H. Procedura PLOT zapala punkt o współrzędnych zawartych w C i B, procedura POINT sprawdza obecność punktu we wsp. C i B sygnalizując to bitem w akumulatorze; procedura FP-TO-A zamienia ten bit na bajt (I — zapalony, 0 — zgaszony).

Ponieważ skok do podprocedury (CALL) także wykorzystuje stos, musiałem adres powrotu z procedury przechowywać w parze DE.

Jeżeli obszar nie jest zamknięty (zapełniamy cały ekran lub obszar przylega do brzegu ekranu), granice obszaru wyznaczają brzegowe linie ekranu (tak jakby było PLOT 0,0: DRAW 255,0: DRAW 0, 175: DRAW — 255,0: DRAW 0, — 175). Procedurę maszynową umieszczamy w pamięci program z listingu 3.

Teraz zamalowanie misia to igraszka, lecz pamiętaj o postawieniu punktu wewnątrz niego.

Marcin Przasnyski

```
10           ORG  60000
20           LD   B,255
30           LD   C,0
40           PUSH BC
50           LD   HL,23677
60           LD   C,(HL)
70           LD   HL,23678
80           LD   B,(HL)
90           LD   H,B
100          LD   L,C
110          JR   START
120
130 PETLA    CALL SPR
140          POP  HL
150          LD   A,H
160          CP   255
170          RET  Z
180 START    LD   C,L
190          LD   B,H
200          INC  B
210          CALL SPR
220          LD   C,L
230          LD   B,H
240          DEC  B
250          CALL SPR
260          LD   C,L
270          DEC  C
280          LD   B,H
290          CALL SPR
300          LD   C,L
310          INC  C
320          LD   B,H
330          JR   PETLA
340
344 SPR      POP  DE
346          PUSH DE
350          LD   A,C
360          CP   0
370          RET  Z
380          CP   255
390          RET  Z
400          LD   A,B
410          CP   0
420          RET  Z
430          CP   175
440          RET  Z
450
460          PUSH BC
470          PUSH HL
475          PUSH DE
480          CALL 8910
490          CALL 11733
495          POP  DE
500          POP  HL
510          POP  BC
520          LD   A,A
530          CP   1
540          RET  Z
545          POP  DE
550
560          PUSH BC
570          PUSH HL
575          PUSH DE
580          CALL 8933
585          POP  DE
590          POP  HL
600          POP  BC
610          PUSH BC
620          RET
```

```
 100 DIM p(2000,2)
 110 LET p=0: LET x=PEEK 23677:
LET y=PEEK 23678
 114 LET xp=x: LET yp=y: GO TO 1
50
 120 GO SUB 200
 130 IF p=0 THEN STOP
 140 LET xp=p(p,1): LET yp=p(p,2
): LET p=p-1
 150 LET x=xp: LET y=yp+1: GO SU
B 200
 160 LET x=xp: LET y=yp-1: GO SU
B 200
 170 LET x=xp-1: LET y=yp: GO SU
B 200
 180 LET x=xp+1: LET y=yp: GO TO
 120
 200 IF x=0 OR y=0 OR x=255 OR y
=175 THEN RETURN
 201 IF POINT (x,y) THEN RETURN
 220 PLOT x,y
 230 LET p=p+1: LET p(p,1)=x: LE
T p(p,2)=y
 240 RETURN
```

```
 10 CLEAR 59999
 20 LET s=0
 30 FOR f=60000 TO 60092
 40 READ a: POKE f,a
 50 LET s=s+a
 60 NEXT f
 70 IF s<>13826 THEN PRINT "Złe
dane": STOP
100 DATA 6,255,14,0,197,33,125
110 DATA 92,78,33,126,92,70,96
120 DATA 105,24,8,205,144,234
130 DATA 225,124,254,255,200,77
140 DATA 68,4,205,144,234,77,68
150 DATA 5,205,144,234,77,13,68
160 DATA 205,144,234,77,12,68
170 DATA 24,225,209,213,121,254
180 DATA 0,200,254,255,200,120
190 DATA 254,0,200,254,175,200
200 DATA 197,229,213,205,206,34
210 DATA 205,213,45,209,225,193
220 DATA 127,254,1,200,209,197
230 DATA 229,213,205,229,34,209
240 DATA 225,193,197,213,201
```

rem jest taka sama jak komend dotyczących magnetofonu:

SAVE • "sd" OPCJA lub LOAD • "sd" OPCJA

OPCJA oznacza LINE numer (tylko dla SAVE), SCREEN$, CODE start, długość lub DATA nazwy tablicy

Program BRIDGE.BAS wczytalibyśmy przez LOAD • „SPECTRUM : GRY.DIR.BRIDGE.BAS". TOS umożliwia korzystanie z programu o nazwie START, który wczytuje się automatycznie i uruchamia po naciśnięciu RESET. Może to być jakikolwiek program nagrany przez SAVE • „START" LINE numer, ale najkorzystniej będzie, jeżeli ułatwimy sobie w ten sposób wyświetlenie katalogu dysku i wczytanie dowolnie wybranego innego programu. Komenda SAVE • nie jest wykonywana natychmiast, jeśli TOS stwierdzi istnienie zbioru o podanej nazwie. Jest to przejawem troski o roztargnionych użytkownikach, którzy nieopatrznie mogliby skasować niewłaściwy program. W każdym z podobnych przypadków system informuje, że napotkał już taką nazwę i żąda ponownego zaakceptowania komendy.

Również przed wykonaniem instrukcji ERASE • "sd" użytkownik ma czas zastanowić się jeszcze raz. Służy ona do usuwania z dysku zbędnych już zbiorów. Oprócz konieczności każdorazowego potwierdzenia wszystkich komend powodujących kasowanie poprzedniego zapisu istnieje jeszcze jeden bardzo wygodny sposób ochrony programów. Jest nim zmiana atrybutów zbioru przez ATTR • "sd" p. Litera p pochodzi od „protected" i oznacza, że zbioru nie można skasować, ani zapisać innego o tej samej nazwie do czasu usunięcia zabezpieczenia przez ATTR • "sd" u. Komenda ATTR • "sd" i utrudnia ingerencję innych osób do naszych programów, ponieważ CAT • nie wyświetla ich nazw. Przeciwne znaczenie ma podanie w komendzie litery v. Do zmiany nazwy zbioru lub katalogu wystarczy podanie LET • "stara nazwa" TO "nowa". Dzięki instrukcji MOVE • "źródło" TO "przeznaczenie" oraz programom zawartym na dysku demonstracyjnym użytkownik nie ma żadnych kłopotów ze sporządzeniem kopii programów lub całej dyskietki.

Istnieje grupa instrukcji ułatwiających poruszanie się w rozbudowanej strukturze katalogów i podkatalogów. Są one odpowiednikiem instrukcji skoku w Basic-u i powodują przemieszczanie się do dowolnie wybranego podkatalogu. Należy do nich instrukcja GOTO • "sd", która zmienia bieżący katalog na inny. Dodatkowe opcje GOTO • "" i GOTO "" zmieniają aktualny katalog na katalog znajdujący się jeden lub dwa poziomy wyżej. Instrukcja GO SUB • "xd" łącznie z • DRAW• odpowiada skokowi do podprogramu w BASIC-u. Aktualne położenie w hierarchicznej strukturze zbiorów podaje LIST•.

Ostatnią grupę instrukcji stanowią instrukcje umożliwiające tworzenie i zarządzanie zbiorami danych o dostępie sekwencyjnym i bezpośrednim. Instrukcja DIM • "sd" zakłada zbiór lub podkatalog na dysku. Do otwarcia i zamknięcia zbioru w wybranym trybie służą

OPEN # • nr kanału; "sd"; tryb dostępu, [; dług. rekordu]

oraz CLOSE # • nr kanału

Odczyt danych ze zbioru umożliwia instrukcja

INPUT • # nr kanału, nazwa zmiennej a zapis w zbiorze wyrażenie PRINT • # nr kanału;

[; AT nr rekordu]

Instrukcja RESTORE • # nr kanału ustawia wskaźnik zbioru na początek

Wszystkie instrukcje TOS mają znaczenie zbliżone do odpowiadających instrukcji występujących w BASIC-u. Bardzo ułatwia to użytkownikowi posługiwanie się nimi. Umiejętne ich wykorzystanie w programie pozwala stworzyć bardzo ciekawe oprogramowanie

Janusz Jarmoch

WRITEST

Nareszcie i KLAN COMMODORE doczekał się własnego programu kontrolującego wpisywanie publikowanych w BAJTKU programów i przeznaczonego dla Commodore 20, 64, 16, 116, PLUS/4 i 128. Od tego numeru BAJTKA wszystkie programy ukazujące się w naszym Klanie będą przedstawiane wraz z kodem kontrolnym co powinno ułatwić życie naszym czytelnikom.

Po wpisaniu programu należy go koniecznie zapisać najpierw na taśmie czy dyskietce i dopiero potem uruchomić. Program automatycznie rozpoznaje z jakim komputerem ma do czynienia. Następnie zaczynamy wpisywać dany program z klanu. Po wpisaniu danej linii i wciśnięciu RETURN lub ENTER, w lewym górnym rogu ekranu ukażą się w negatywie (rewersie) dwa znaki stanowiące kod kontrolny. Kod ten należy porównac z kodem podanym przy wydruku programu (listingu). Jeżeli kody te różnią się należy dokładnie sprawdzić daną linię programu — oznacza to, ze została ona wpisana z błędem.

na podstawie COMPUTE! s GAZETTE.

Jan Jasiński
COMMODORE CLAN KOMODA

```
BB  100 POKE53280,0:POKE53281,0:PRINTCHR$(147)CHR$(5)
BF  105 BN=43:BW=44:WSK=PEEK(772)+256*PEEK(773)
EB  110 PRINT" WRITEST":PRINT
F0  115 PRINT" JAN JASINSKI":PRINT
4E  120 PRINT" (C) 1987 KRAKOW":PRINT
CA  125 PRINT" COMMODORE ";:IFWSK=42364THENPRINT"C64"
73  130 IFWSK=50556THENPRINT"VIC20"
7F  135 IFWSK=35158THENGRAPHICCLR:PRINT"+4/16"
55  140 IFWSK=17165THENBN=45:BW=46:GRAPHICCLR:PRINT"P
        C128"
72  145 AP=(PEEK(BN)+256*PEEK(BW))+6:ADR=AP
78  150 FORI=0TO166:READBAJT:POKEADR,BAJT:ADR=ADR+1:S
        UMA=SUMA+BAJT:NEXT
34  155 IFSUMA<>20312THENPRINT"SPRAWDZ LINIE 200-280"
        :END
F7  160 FORI=1TO5:READPR,NR,WR:NA=AP+PR:BS=INT(NA/256
        ):BM=NA-(256*BS)
E7  165 SUMA=SUMA+PR+NR+WR:POKEAP+NR,BM:POKEAP+WR,BS:
        NEXT
0C  170 IFSUMA<>21796THENPRINT"SPRAWDZ LINIE 280-290"
        :END
11  175 POKEAP+149,PEEK(772):POKEAP+150,PEEK(773)
99  180 IFWSK=17165THENPOKEAP+14,22:POKEAP+18,23:POKE
        AP+29,224:POKEAP+139,224
57  185 PRINTCHR$(147)"WRITEST JEST GOTOWY DO PRACY..
        .":SYS AP
35  190 POKEBW,PEEK(BW)+1:POKE(PEEK(BN)+256*PEEK(BW))
        -1,0:NEW
FE  195 :
73  200 DATA 120,169,073,141,004,003,169,003,141,005
F2  205 DATA 003,088,096,165,020,133,167,165,021,133
A7  210 DATA 168,169,000,141,000,255,162,031,181,199
14  215 DATA 157,227,003,202,016,248,169,019,032,210
7F  220 DATA 255,169,018,032,210,255,160,000,132,180
0F  225 DATA 132,176,136,230,180,200,185,000,002,240
91  230 DATA 046,201,034,208,008,072,165,176,073,255
C5  235 DATA 133,176,104,072,201,032,208,007,165,176
5E  240 DATA 208,003,104,208,226,104,166,180,024,165
20  245 DATA 167,121,000,002,133,167,165,168,105,000
F9  250 DATA 133,168,202,208,239,240,202,165,167,069
0E  255 DATA 168,072,041,015,168,185,211,003,032,210
D1  260 DATA 255,104,074,074,074,074,168,185,211,003
80  265 DATA 032,210,255,162,031,189,227,003,149,199
CF  270 DATA 202,016,248,169,146,032,210,255,076,086
96  275 DATA 137,048,049,050,051,052,053,054,055,056
AF  280 DATA 057,065,066,067,068,069,070,013,002,007
25  285 DATA 167,031,032,151,116,117,151,128,129,167
5B  290 DATA 136,137
```

PCHEŁKA $01

Podana obok „pchełka" umożliwia ustawianie kursora we wskazanym miejscu ekranu 40-znakowego (programik ten jest przeznaczony dla C–64). Zmienne X i Y określają miejsce do którego zostanie przesunięty kursor — X oznacza kolumnę, Y natomiast wiersz. Procedura zlokalizowana jest od adresu 49152. Format w jakim nalezy przesyłać współrzędne :SYS 49152,X,Y Programik ten nadaje się znakomicie do prostych gier i został zaczerpnięty z książki GAMES COMMODORE PLAY.

Opracował (kd)

```
71  100 REM   *** PCHELKA  $01 ***
38  105 :
49  110 REM     PHILIP W. DENNIS
F8  115 :
DE  125 FOR I=0 TO 24:READ C
B2  130 POKE 49152+I,C
2A  135 NEXT
F9  140 :
26  145 NEW
BA  150 :
60  155 DATA 032,016,192,132,253
86  160 DATA 032,016,192,152,170
53  165 DATA 164,253,024,076,240
E0  170 DATA 255,032,253,174,032
9D  175 DATA 158,173,076,170,177
```

Turbo 64 Board

Gdy chociaż jeden raz usiądziemy przed nowoczesnym komputerem 16-bitowym typu Apple IIgs, Amiga czy też Atari ST i wykorzystamy ich moc obliczeniową i szybkość przetwarzania tych maszyn, to po powrocie do naszego poczciwego Commodore 64 mamy wrażenie jakby czas stanął w miejscu. Wszystkie programy wykonywane są dużo wolniej, a do niektórych zagadnień nie zabieramy się w ogóle wiedząc, że nie starczy nam na to pamięci, czy że po prostu program będzie chodził za wolno.

W sukurs takim użytkownikom przyszła szwajcarska firma Swisscomp — jeden z dostawców ciekawych rozwiązan sprzętowych do produktów firmy Commodore. Ostatnią ofertą jest moduł o nazwie Turbo 64; firma przyznała się, ze karta ta powstała w Europie, a dopiero potem wyemigrowała za ocean (zwykle nowinki techniczne przybywały drogą odwrotną). O niej właśnie będzie dzisiaj mowa.

Karta ta może pracować z częstotliwością zegara 1 lub 4 MHz (wartośc wybierana przez uzytkownika w sposób sprzętowy lub programowy). Posiada ona wbudowany 16-bitowy procesor 65816 (taki sam procesor jak w Apple IIgs) dzięki ktoremu może pracować w trybie 16-bitowym czy tez emulować 8-bitowy procesor 6510. W trybie 6510/4 MHz wszystkie programy na C–64 wykonywane są do czterech razy szybciej. Tryb pracy, oparty o procesor 16-bitowy daje możliwość pisania programów działających jeszcze szybciej, a więc otwiera nam pole do nowych zastosowań. Kartę podłączamy tak samo jak inne „klasyczne" moduły — do portu rozszerzającego (expansion port). Oprócz nowej jednostki centralnej 65816, karta Turbo 64 posiada: 64KB pamięci RAM (wykonanej w technologii CMOS) z zasilaniem bateryjnym, system operacyjny emulujący pracę procesora 6510 zawarty w pamięci EPROM oraz możliwości dalszego sprzętowego rozbudowywania naszego komputera. Cena takiej przyjemności — 189 dolarów amerykańskich — nie jest niska, ale jeśli wziąć pod uwagę, ze w ten sposób praktycznie otrzymujemy na karcie kompletny komputer 16-bitowy (oprócz operacji wejścia/wyjścia, wykonywanych tak jak w C–64) i zachowujemy zgodność programową z C–64, to poniesienie takiego kosztu wydaje się chyba godne rozważenia.

Karta Turbo 64 pracuje ze standardowym C–64, C–64C i SX–64. Może być również używana w C–

128, pracującym w trybie C–64, choć pojawiają się wtedy pewne problemy natury technicznej, karta bowiem generuje zakłócenia radiowe (RFI), uniemożliwiając normalną pracę C–128. Dodatkowe ekranowanie komputera jest całkowicie wystarczające i skutecznie eliminuje tę wadę. Turbo 64 nie współpracuje niestety z C–128 i 128D w trybie 128.

Turbo 64 z wyglądu jest większa niż standardowy moduł, a jej wymiary zbliżone są do wymiarów karty Z–80, oferowanej kiedyś przez firmę Commodore. Wadą tej karty jest to, że wszystkie układy elektroniczne nie są niczym osłonięte od otoczenia, a więc gwałtownie wzrasta niebezpieczeństwo przebicia elektrostatycznego, a jak wiemy układy elektroniczne — szczególnie wykonane w technologii CMOS — nie wytrzymują dużych potencjałów. Jedyną możliwością, ze względów estetycznych i ochronnych, pozostaje obudowanie karty sposobem chałupniczym. Karta ta, oprócz już na wstępie wymienionych układów elektronicznych, posiada 6 przełączników, diodę elektroluminescencyjną (LED), trzy potencjometry i klawisz RESET. Wszystko to służy do kontroli różnych funkcji i trybów pracy karty, niestety pobieżnie opisanych w dołączonej instrukcji obsługi. Jak wspomniałem na wstępie, pamięć RAM posiada podtrzymanie bateryjne. Teoretycznie oznacza to, że cokolwiek znajdowało się w tej pamięci, pozostanie w niej nawet po wyłączeniu komputera. Praktycznie jednak po wyłączeniu i ponownym włączeniu komputera wskaźniki systemu operacyjnego ustawiane są na wartościach takich jak przy normalnym uruchomieniu komputera; dlatego też, jeżeli nasz program pisany był w BASIC-u musimy wskaźnikom tym przypisać te wartości jakie były ustalone przed wyłączeniem komputera. Karta jest też przystosowana do podłączenia (mającej powstać również w Swisscomp) modułu rozszerzającego pamięć do wartości 1 MB. Jest to możliwe ponieważ przestrzeń adresowa procesora 65816 wynosi 16 MB. Niestety pamięć ta nie będzie dostępna dla C–64 pracującego w oparciu o emulator procesora 6510.

Jak wspomniałem istalacja karty jest bardzo prosta. Pewien problem pojawia się w momencie gdy chcemy do komputera dołączyć więcej kart — firma rozwiązała go nieco mniej elegancko, gdyż niezbędne jest wtedy dokupienie odpowiedniego „rozgałęziacza". Następnie przez odpowiednie ustawienie przełączników wybieramy tryb pracy i karta jest gotowa do użytku. Na początku użytkownikom tej karty może wydawać się, że Trubo 64 przyspiesza jedynie działanie starej jednostki centralnej lecz jest to złudzenie — w rzeczywistości jest ona całkowicie wyłączona. Procesor 65816 w pełni emuluje 6510 za pomocą kombinacji programowej i sprzętowej. Firma zapewnia, że bezpośrednie instrukcje POKE, PEEK trafiają tam gdzie powinny trafić. Również całkowita emulacja systemu operacyjnego C–64 (KERNAL) ma miejsce w pamięci RAM obsługiwanej przez 65816. Właśnie te cechy umożliwiają pełną zgodność z C–64 w trybie 1 MHz jak i w 4 MHz.

Pomimo dość wysokiej ceny, wydaje się, że Turbo 64 stanowi ciekawe i interesujące novum dające użytkownikowi C–64 szereg nowych możliwości do wykorzystania — miejmy nadzieję, że producent nie zapomni również o oprogramowaniu przeznaczonym dla tego nowego procesora zainstalowanego na karcie.

na podstawie „COMPUTER SHOPPER" 8/87

Dominik Falkowski

Złośliwy chochlik nadal nęka KLAN COMMODORE. Tym razem wkradł się do artykułu „INSTRUKCJE $SHAPE I GHSAPE publikowanym w numerze 10 BAJTKA — w linii 180 programu zawarta jest instrukcja SLEEP 5, która nie jest implementowana w Commodore PLUS/4 i C-16/116. Zastąpić ją można zwykłą pętlą opóźniającą np.

180 FOR I=1 TO 3000 : NEXT I

gdyż jej zadaniem jest zatrzymanie obrazu na ekranie przez 5 sekund. Za błąd serdecznie Czytelników przepraszam.

(k.d.)

WPROWADZANIE FUNKCJI DLA -C-64-

Jedną z większych wad wersji BASIC V2.0 Commodore 64 jest niemożność wprowadzania za pomocą instrukcji INPUT własnych funkcji czy wyrażeń algebraicznych.

Jak wadę tę zlikwidować? Bardzo prosto. Wpisz poniżej przedstawiony program.

Powyższej wady pozbawiony jest np. komputer ZX Spectrum w którym ciąg instrukcji:

INPUT A$: X = VAL A$

umożliwia obliczenie danego wyrażenia. Dla C-64 równoważnym ciągiem będzie

INPUT A$: £VAL X,A$

Procedura ta, napisana w języku wewnętrznym zlokalizowana jest w obszarze 4 dodatkowych KB pamięci od adresu 49152 ($C000). Po uruchomieniu i wykonaniu program wczytujący kasuje się samoczynnie. Procedura ta jest włączona w istniejącą pętlę interpretera, tak więc posługiwanie się nią nie przeszkadza równoległemu korzystaniu z innych rozszerzeń takich jak np. TURBO pod warunkiem, że obszary przez nie zajmowane nie kolidują ze sobą, oraz, że procedura tu omawiana będzie wczytana i uruchomiona jako ostatnia. Wynika to z faktu, że większość rozszerzeń wykorzystuje (i zmienia) wektor zawarty w komórkach 776 i 777, bez uwzględnienia wartości dotychczasowych. Wady tej pozbawiony jest omawiany tu program. Po wprowadzeniu wzoru danego wyrażenia (np. 2+2*5) za pomocą instrukcji INPUT, jego wartość jest obliczana poprzez £VAL X,A$ a wynik jest przypisywany zmiennej oznaczonej w przykładzie jako X.

Krystian Łojewski

```
100 REM *** WPROWADZANIE FUNKCJI ***
101 :
102 :
105 A=49152
115 B=A:FORI=0TO33:S=0:FORJ=0TO9:READX:S=S+X
125 POKEB,X:B=B+1
135 NEXTJ:READX
145 IFS<>X THENPRINT"ERROR W LINII":165+I:END
155 NEXT I
156 :
157 SYS 49152
158 NEW
159 :
165 DATA 174,008,003,172,009,003,224,032,208,005,0838
166 DATA 192,192,208,001,096,142,062,193,140,063,1289
167 DATA 193,169,032,141,008,003,169,192,141,009,1057
168 DATA 003,096,165,122,072,165,123,072,032,115,0965
169 DATA 000,201,092,208,007,032,115,000,201,197,1053
170 DATA 240,009,104,133,123,104,133,122,108,062,1138
171 DATA 193,104,104,032,115,000,201,044,240,003,1036
172 DATA 076,008,175,032,115,000,165,122,133,251,1077
173 DATA 165,123,133,252,032,138,173,165,122,133,1436
174 DATA 253,165,123,133,254,032,253,174,032,158,1577
175 DATA 173,160,002,177,100,153,064,193,136,016,1174
176 DATA 248,160,000,162,000,177,251,157,083,193,1431
177 DATA 232,230,251,208,002,230,252,165,251,197,2018
178 DATA 253,208,238,169,178,157,083,193,173,065,1717
179 DATA 193,133,251,173,066,193,133,252,232,138,1764
180 DATA 024,105,083,133,253,169,000,105,193,133,1198
181 DATA 254,173,064,193,208,003,076,072,178,173,1394
182 DATA 064,193,240,106,169,165,141,067,193,162,1500
183 DATA 000,160,000,189,047,161,048,007,153,069,0834
184 DATA 193,200,232,208,244,041,127,153,069,193,1660
185 DATA 200,140,068,193,136,173,064,193,205,069,1440
186 DATA 193,144,046,177,251,217,069,193,208,039,1537
187 DATA 136,016,246,173,067,193,200,145,253,173,1602
188 DATA 068,193,024,101,251,133,251,144,002,230,1397
189 DATA 252,230,253,208,002,230,254,056,173,064,1722
190 DATA 193,237,068,193,141,064,193,176,166,238,1669
191 DATA 067,193,232,224,072,208,170,160,000,177,1503
192 DATA 251,145,253,169,001,141,068,193,206,208,1637
193 DATA 160,007,185,075,193,145,253,136,016,248,1418
194 DATA 165,122,133,251,165,123,133,252,169,082,1595
195 DATA 133,122,169,193,133,123,076,174,167,165,1455
196 DATA 251,133,122,165,252,133,123,096,228,167,1670
197 DATA 000,227,159,194,001,065,084,078,080,000,0888
198 DATA 000,058,158,052,057,052,054,049,058,068,0606
```

NUMERATOR

Poniższy program, choć dość krótki, może być bardzo przydatny wszystkim posiadaczom stacji dysków — umożliwia on wpisanie do katalogu dyskietki (directory) numeru składającego się z 5 znaków.

Podczas wyświetlania katalogu za pomocą np. LOAD"$",8 czy DIRECTORY, w miejscu identyfikatora (ID) oraz znaków określających numer systemu operacyjnego (2A) ukaże się przypisany przez nas numer. Stanowczo odradzam wprowadzanie takich zmian na dyskietki oryginalne (firmowe, z nagranym już programem), gdyż może to spowodować późniejsze trudności z jego wczytywaniem, jeśli program kontroluje te pola np. w procedurze obsługi (zamierzonego) błędu.

Wprowadzenie zmian do programu polecałbym jedynie użytkownikom zaawansowanym, gdyż wykorzystuje on rozkazy bezpośrednie systemu DOS; źle wpisana przeróbka może w ekstremalnych wypadkach zniszczyć lub na trwałe zablokować dane zapisane na dyskietce. Program można uruchomić na każdym komputerze Commodore z przyłączoną stacją dysków 1540, 1541, 1570, 1571 czy 1572.

Klaudiusz Dybowski

```
02  100 REM  ***  NUMERATOR  ***
3B  105 :
49  110 REM   KLAUDIUSZ DYBOWSKI
FB  115 :
78  120 :
C3  125 PRINT CHR$(147) : PRINT : PRINT
5C  130 PRINT"  WLOZ DYSKIETKE DO STACJI I
        WCISNIJ RETURN."
99  135 GET A$:IF A$<>CHR$(13) GOTO 135
E2  140 OPEN 15,8,15
44  145 OPEN 5,8,5,"#"
74  150 PRINT#15,"U1";5;0;18;0
D8  155 PRINT#15,"B-P";5;162
B9  160 FOR X=1 TO 5
2B  165 GET#5,B$:IF B$="" THEN B$=CHR$(0)
29  170 C$=C$+B$
37  175 NEXT : PRINT : PRINT
80  180 PRINT"  STARY ZAPIS : ";C$ : PRINT
        : PRINT
2F  185 INPUT"  WPISZ NOWY NUMER (5 ZNAKOW)
        : ";D$
8D  190 A=LEN(D$)
F6  195 IF A<>5 THENPRINT CHR$(145)CHR$(145
        ) : GOTO 185
0F  200 PRINT#15,"B-P";5;162
09  205 PRINT#5,D$
30  210 PRINT#15,"U2";5;0;18;0
AB  215 CLOSE 5
C9  220 PRINT#15,"I"
EB  225 CLOSE15
64  230 END
```

PRZENIEŚĆ OBRAZ



Choć poszczególne modele Commodore różnią się od siebie, to mają one także jedną cechę wspólną — jest nią grafika o rozdzielczości 320*200 punktów. Dla entuzjastów grafiki mam więc coś ekstra — sposób przenoszenia obrazów graficznych pomiędzy modelami C-64, C-16 /116/PLUS4 i C-128.

Przenoszenie to jest możliwe w każdym kierunku — można zaprojektować rysunek na C-128 i przenieść go następnie do C-16 czy C-64 czy też odwrotnie — rysunki wykonane np. za pomocą programu HI EDDI na C-64 można obejrzeć bez problemu na C-16 czy C-128. Opisane programy odnoszą się wyłącznie do standardowego trybu graficznego wysokiej rozdzielczości (HIRES) i nie dotyczą trybu wielokolorowego (multicolor).

Przenoszenie rysunków z komputera na komputer odbywa się poprzez ich zapis na taśmie lub dyskietce. Programy o numerach linii zaczynających się na 1 działają z magnetofonem natomiast programy o numerach zaczynających się na 8 ze stacją dysków.

```
42   10 REM ** LTAPE.128 **
D4   11 :
A7   12 GRAPHIC 1,1:GRAPHIC 0
8D   13 A=A+1:IF A=2 THEN 15
02   14 IFA=1THENLOAD"RYS.",1,1
F7   15 GRAPHIC 1
65   16 GETKEY A$
36   17 GRAPHIC 0

0E   10 REM ** LTAPE.116/+4 **
D4   11 :
EC   12 A=A+1:IF A=2 THEN 14
79   13 IFA=1THENGRAPHIC1,1:GRAPHI
        C0:LOAD"RYS.".1.1
06   14 GRAPHIC 1
85   15 GETKEY A$
46   16 GRAPHIC 0

CB   10 REM ** LTAPE.64 **
D4   11 :
DD   12 A=A+1:IF A=2 THEN 15
4B   13 FORX=8192TO16383:POKEX,0:N
        EXT
02   14 IFA=1THENLOAD"RYS.",1,1
27   15 FORX=1024TO2023:POKEX,128:
        NEXT
20   16 POKE53272,PEEK(53272)OR8
80   17 POKE53265,PEEK(53265)OR32
05   18 GOTO18
```

Program P. MAKER służy jedynie do wykreślenia rysunku demonstracyjnego i może być wykorzystany w C-16/ 116/ PLUS4 oraz C-128. W wypadku C-64 proponowałbym spróbować przenieść rysunek wykonany za pomocą programu HI EDDI lub podobnego wykorzystującego jako ekran graficzny obszar pamięci od adresu 8192 ($2000).

W modelach C-16/116/PLUS4 i C-128 wykonanie GRAPHIC 1 powoduje zarezerwowanie pamięci dla ekranu graficznego od adresu 8192 ($2000) do 16383 ($3FFF). Na tym też oparta jest cała idea — jeżeli dowolny rysunek będzie zapisany w tym

```
E5   80 REM ** LDISK.128 **
89   81 :
7F   82 GRAPHIC 1,1
EE   83 A=A+1:IF A=2 THEN 85
B1   84 IFA=1THENBLOAD"RYS.",B0,P8
        192
CA   85 GETKEY A$
9B   86 GRAPHIC 0

77   80 REM ** LDISK.116/+4 **
89   81 :
BB   82 GRAPHIC 1
EE   83 A=A+1:IF A=2 THEN 85
1D   84 IFA=1THENSCNCLR:LOAD"RYS."
        ,8,1
CA   85 GETKEY A$
9B   86 GRAPHIC 0

B2   80 REM ** LDISK.64 **
B9   81 :
8D   82 A=A+1:IF A=2 THEN 88
6C   83 POKE 53272,PEEK(53272)OR8
4F   84 POKE 53265,PEEK(53265)OR32
AC   85 FORX=1024TO2023:POKEX,128:
        NEXT
E0   86 FORX=8192TO16383:POKEX,0:N
        EXT
E2   87 IF A=1 THENLOAD"RYS.",8,1
ED   88 GOTO88
```

obszarze pamięci, to można go bez problemu przenieść na jakikolwiek model Commodore. Dotyczy to oczywiście nie tylko rysunków własnych wykonanych w BASIC lecz także rysunków wykonanych za pomocą programów takich jak HI EDDI PLUS (C-64) czy GIGA CAD (C-64). Generalnie rzecz biorąc, każdy rysunek zapisany w tym obszarze pamięci powinien dać się przenieść na dowolny typ Commodore.

Jak sprawdzić, czy rysunek zapisany na dyskietce tworzony był w tym obszarze pamięci czy nie? Bardzo prosto. Do odczytania adresu posłuż się programem ADRES. Po jego uruchomieniu wpisz dokładnie nazwę rysunku zapisanego na dysku. Jeżeli w wyniku otrzymasz na ekranie komunikat ADRES = 8192 oznacza to, że rysunek taki może być wczytywany i przenoszony. Korzyści płynące z ta-

```
C7   20 REM *** P.MAKER ***
65   21 :
1B   22 GRAPHIC 1,1
C5   23 :
80   24 DRAW 1,000,100 TO 319,100
0E   25 DRAW 1,160,000 TO 160,199
5E   26 FOR X=10 TO 90 STEP 3
21   27 CIRCLE 1,160,100,70,X
2A   28 CIRCLE 1,160,100,X,50
F2   29 NEXT X
BC   30 CHAR 1,9,24,"RYSUNEK DEMON
        STRACYJNY"
36   31 :
10   32 FOR D=1 TO 1000:NEXT D
9C   33 :
47   34 GRAPHIC 0

E2   80 REM *** ADRES ***
89   81 :
34   82 INPUT"FILENAME";F$
C4   83 OPEN1,8,5,F$+",R"
3E   84 Z$=CHR$(0)
C9   85 GET#1,A$,B$
19   86 AD=ASC(A$+Z$)+256*ASC(B$+Z
        $)
24   87 PRINT"ADRES = ";AD
8E   88 CLOSE1
```

kiej „wymiany" są widoczne od razu. Możemy zaprojektować np. planszę tytułową do naszej gry dla C-64 czy programu użytkowego korzystając z lepszej wersji BASIC powiedzmy Commodore PLUS/4, przenieść na C-128 rysunek, który wykonywany był gdy posiadałeś C-64 czy C-16 itp. Dla każdego coś się znajdzie.

ZAPIS RYSUNKOW

Gdy nasz rysunek zostanie już wykonany i chcemy go zapisać na taśmie czy dyskietce należy skorzystac z pomocy instrukcji MONITOR (C-16/116/ PLUS4/128) lub odpowiednio zrekonfigurować pamięć i zapisać jej wydzielony obszar na dyskietce (C-64). W pierwszym wypadku zapisujemy obraz za pomocą:

S"RYS.", 01 2000 4000
(zapis na kasecie)
lub
S"RYS.", 08 2000 4000
(zapis na dyskietce)

Dla C-64 zapis obrazu jest możliwy po wykonaniu (w trybie ekranowym).

POKE43, 0: POKE44, 32. POKE45,1: POKE46,64: SAVE "RYS.",1,1

Jeżeli rysunek ma być zapisany na dyskietce ostatnią instrukcję należy zmodyfikować:

...: SAVE "RYS.",8,1

UWAGA. W wypadku komputerów C-128D może się zdarzyć, że przenoszenie (wczytywanie rysunkow z kasety spowoduje pewne problemy ze względu na specyficzną konstrukcję tego modelu. Uwaga ta, zgodnie z podręcznikiem COMMODORE 128 PROGRAMMER'S REFERENCE GUIDE może także dotyczyć modelu C-128.

WCZYTYWANIE RYSUNKU

Jeżeli rysunek będzie wczytywany ze stacji dysków, to jednocześnie możemy obserwować na ekranie jego wykreślanie. Podczas wczytywania z kasety ekran jest wyłączany, tak więc efekt będzie widoczny dopiero po zakończeniu wczytywania.

Klaudiusz Dybowski

EDYTOR BASIC-a

Przy przepisywaniu programów w BASIC-u, szczególnie zawierających dużo liczb w instrukcjach DATA, nie sposób ustrzec się błędów. Aby uniknąć żmudnego wyszukiwania popełnionych omyłek wszystkie programy w BASIC-u będą zamieszczane wraz z kodami kontrolnymi. Poniższy program „Edytor BASIC-a" umożliwia obliczanie i kontrolę kodów wpisywanych linii.

Najpierw należy dokładnie przepisać zamieszczony tu wydruk i zapisać go na kasecie lub dysku. Przystępując do wpisywania dowolnego programu z naszego pisma trzeba wczytać i uruchomić „Edytor BASIC-a". Następnie należy wpisywać linie programu. Po wpisaniu linii i naciśnięciu RETURN pojawia się ona w dolnej części ekranu wraz z obliczonym kodem kontrolnym. Jeżeli wyświetlony kod jest taki sam jak wydrukowany przed numerem linii, można wpisywać następną linię. Jeśli kody są różne, to ponowne naciśnięcie RETURN powoduje wyświetlenie wpisanej linii w górnej części ekranu i umożliwia dokonanie poprawek. Wpisanie samego numeru linii powoduje wymazanie z pamięci komputera linii programu o tym numerze. Naciśnięcie RETURN wywołuje zawsze ostatnio wpisaną linię. Aby wywołać inną, wcześniej wpisaną linię należy napisać numer tej linii z gwiazdką przed nim (np. *140) i nacisnąć RETURN.

Po wpisaniu całego programu trzeba przerwać pracę „Edytora" naciśnięciem klawisza BREAK lub RESET. Następnie w celu usunięcia „Edytora" zapisujemy program na taśmie instrukcją LIST "C:",0,31999 lub na dysku instrukcją LIST "D:nazwa",0,31999. Teraz trzeba wpisać rozkaz NEW i odczytać program instrukcją ENTER "C:" lub ENTER "D:nazwa". Po tej operacji w pamięci komputera znajduje się wpisany program bez „Edytora BASIC-a" i można go już ostatecznie zapisać na nośnik.

```
32000 REM EDYTOR BASIC-A
32010 REM wersja 1.0 dla "Bajtka"
32020 CLR :DIM LINIA$(120):CLOSE #2:CL
OSE #3
32030 OPEN #2,4,0,"E:":OPEN #3,5,0,"E:
"
32040 ? "↰":POSITION 11,1:? "EDYTOR B
ASIC-A"
32050 TRAP 32040:POSITION 2,3:? "Wpisz
 linie programu"
32060 POSITION 1,4:? " ":INPUT #2;LINI
A$:IF LINIA$="" THEN POSITION 2,4:LIST
 B:GOTO 32060
32070 IF LINIA$(1,1)="*" THEN B=VAL(LI
NIA$(2,LEN(LINIA$))):POSITION 2,4:LIST
 B:GOTO 32060
32080 POSITION 2,10:? "CONT"
32090 B=VAL(LINIA$):POSITION 1,3:? " "
;
32100 POKE 842,13:STOP
32110 POKE 842,12
32120 ? "↰":POSITION 11,1:? "EDYTOR B
ASIC-A":POSITION 2,15:LIST B
32130 C=0:ODP=C
32140 POSITION 2,16:INPUT #3;LINIA$:IF
 LINIA$="" THEN ? "LINIA ";B;" USUNIET
A":GOTO 32050
32150 FOR D=1 TO LEN(LINIA$):C=C+1:ODP
=ODP+(C*ASC(LINIA$(D,D))):NEXT D
32160 KOD=INT(ODP/676)
32170 KOD=ODP-KOD*676
32180 KODS=INT(KOD/26)
32190 KODM=KOD-(KODS*26)+65
32200 KODS=KODS+65
32210 POSITION 0,16:? CHR$(KODS);CHR$(
KODM)
32220 POSITION 2,13:? "Jezeli kod sie
nie zgadza, nacisnij  RETURN  i popr
aw linie.":GOTO 32050
```

TAJEMNICE ATARI (5)

Zanim przystąpimy do zdradzania sekretow mamy prośbę do tych wszystkich czytelników Bajtka, którzy sami odkryli jakies „tajemnice Atari", aby do nas napisali. Czekamy na listy z takim dopiskiem na kopercie. A oto kolejne „klucze do zagadek".

THE PRICE MAGIC (LEVEL 9 COMPUTING) to gra tekstowo-obrazkowa w języku angielskim. Gry tego typu sprawiają zwykle najwięcej kłopotów. Raz może to być brak jakiegoś słowa, innym razem odrobiny wyobraźni. Czasami jednak może się zdarzyć, że mała podpowiedź pociągnie za sobą lawinę znakomitych pomysłów. W grze „Price of Magic" szczególną rolę odgrywają czary i zaklęcia. W przypadku połączenia ich z odpowiednimi przedmiotami końcowy sukces okaże się murowany. A tak wygląda recepta:

a) kryształowa kula (Crystall Ball) — zaklęcie ESP
b) księga czarow (Grimoire) — zaklęcie MAD
c) wahadło (Pendulum) — zaklęcie DOW
d) pryzmat (Prism) — zaklęcie XAM
e) miotła (Broom) — zaklęcie FLY
f) trąbka (Trumpet) — zaklęcie BOM
g) soczewka ze skalenia (Feldsparlens) — zaklęcie SEE
h) popioły (Ashes) — zaklęcie ZAP
i) berło (Staff) — HYP
j) krzyż (Cross) — zaklęcie DET
k) świeca (Candle) — zaklęcie SPY
l) topór (Axe) — zaklęcie KIL
m) koło (Wheel) — zaklęcie DED
n) skrzynka (Box) — zaklęcie IBM
o) pazur (Claw) — zaklęcie SAN
p) waleriana (Valerian) — zaklęcie FIX
r) ognisko (Focus) — zaklęcie SPELL
s) lustro (Mirror) — zaklęcie LIT/ZEN

FIGHT NIGHT firma Accolade to jeden z tegorocznych przebojow giełdy komputerowej przy ulicy Grzybowskiej. Gra zupełnie inna niż „Price of Magic". W tym przypadku posiadacz ATARI ma szanse stoczenia szeregu fascynujących pojedynków bokserskich. Walkę rozpoczyna się od eliminacji, a przy pewnej dozie umiejętności i szczęścia może dojść do wielkiego finału i rozgrywki o tytuł mistrza świata zarazem. Znakomita grafika tego programu bardzo podnosi atrakcyjność zabawy. Przed rozpoczęciem gry warto jednak wiedzieć, że każdy z potencjalnych przeciwników ma w swoim repertuarze jedno uderzenie niezgodne z przepisami bokserskiego fair play, a także każdy z nich popełnia błąd w chwili przygotowywania się do zadania nieczystego ciosu. Po prostu przed wykonaniem uderzenia jego ruchy na ringu stają się wolniejsze. Fakt ten można obrócić na swoją korzyść. Należy natychmiast cofnąć się, a następnie błyskawicznie zaatakować, tak by nie stracić pola walki. Jeżeli komuś nie odpowiada tempo gry wystarczy wcisnąc klawisz ESC i trzymać go, a natychmiast ruchy przeciwnika staną się wolniejsze. Podczas zabawy trzeba pamiętać, że jedyną receptą na zwycięstwo jest nieustanny atak. Sama tylko obrona jest równoznaczna z wyrażeniem zgody na porażkę.

GAUNTLET to program, który zrobił niezwykłą karierę w Wielkiej Brytanii w 1987 roku, wypuszczony przez firmę US Gold. Uczestnik tej gry może dowolnie wcielić się w jedną z następujących postaci: Merlin'a Wizzard, Thor'a Warrior, Thyr'a Valkirie czy Questor'a Elf. Następny krok to wyjście na przeciw rzuconemu wyzwaniu (rękawicy — GAUNTLET) i wędrówka w nieznane. Pokonanie 512 poziomów w pierwszej części gry i następnych 512 w drugiej to zadanie bardzo trudne i ponad siły przeciętnego „gracza ATARI". Istnieje jednak sposób, aby niewykonalne uczynić wykonalnym. Można przedłużyć życie głównego bohatera. W tym celu (w przypadku jednego tylko gracza) należy podłączyc do komputera drugi Joystick i ustawić grę dla dwóch zawodników. W momencie gdy wskaźnik zdrowia (Health) uczestnika zabawy będzie bliski zeru należy WCISNĄĆ fire w drugim Joysticku. Wtedy pojawi się kolejny bohater. Jeżeli z kolei on będzie bliski śmierci trzeba ponownie włączyć FIRE w Joystick'u numer jeden. Czynności te można powtarzać wielokrotnie, a podany sposób postępowania umożliwi dowolne wydłużenie czasu gry.

BRUCE LEE — od tego programu niewątpliwie zaczynała większość młodych czytelników Bajtka. To, że przejście poszczególnych etapów jest wielce skomplikowane, a kolejne „życia" traci się w oka mgnieniu, wiedzą pewnie wszyscy. Ale tylko niektórzy odkryli tajemnicę zwiększania liczby możliwych niepowodzeń, a przez to wydłużania czasu gry i wzrostu szans dotarcia do celu. Teraz sekret ten stanie się prawdą oczywistą. Pierwszy krok na drodze do sukcesu to odnalezienie komnaty, w której jest małe okrągłe słońce z rysunkiem przypominającym twarz. Drugi krok to zabranie „słoneczka" ze sobą. Ruch ten spowoduje wzrost liczby „istnień" do dyspozycji gracza o jedno. Opuszczenie komnaty, a następnie powrót do niej wywołuje ponowne pojawienie się wizerunku słońca. W ten sposób powstaje kolejna możliwość zwiększenia wysokości ilości możliwych niepowodzeń. Opisane czynności można powtarzać aż do momentu, w którym liczba „istnień" przysługujących graczowi osiągnie maximum.

Sergiusz Piotrowski
Tomasz Mazur

KTÓRY LEPSZY?

Przedstawiliśmy w czterech kolejnych numerach „Bajtka" różne interpretery BASIC-a dla komputerów Atari. Teraz czas na ich porównanie.

Dwa z nich są dostępne na kasetach lub dyskietkach, a dwa jako moduły ROM (cartridge). Determinuje to wstępnie zakres ich użycia. Dla posiadacza stacji dysków rodzaj nośnika — dyskietka czy cartridge — nie stanowi większej różnicy. Natomiast osoby korzystające wyłącznie z magnetofonu będą preferowały moduły ROM ze względu na długi czas odczytu z kasety.

SEKUNDY, SEKUNDY...

Jednymi z najistotniejszych cech każdego interpretera są szybkość pracy i dokładność wykonywania obliczeń arytmetycznych. Parametry te zostały sprawdzone przy pomocy dziewięciu programów testujących, których wydruki zamieszczone są obok. Osiem z nich służy do badania szybkości wykonywania podstawowych instrukcji, zaś test dziewiąty pozwala na określenie dokładności z jaką komputer przeprowadza obliczenia numeryczne. Wyniki prób zostały ujęte w tabeli, która zawiera czasy wykonywania poszczególnych testów (w sekundach) oraz osiągniętą dokładność. FAST oznacza pracę interpretera w trybie szybkim, a M zmodyfikowanie programu w celu dostosowania do specyficznej struktury języka.

Z tabeli jasno wynika, że wszystkie interpretery są szybsze od Atari BASIC. Najlepszy jest tu Turbo BASIC, ale niewiele ustępuje mu BASIC XE. Zadziwiająca jest także szybkość Microsoft BASIC, lecz uzyskana jest kosztem dokładności mniejszej o rząd wielkości. Najgorzej w tym porównaniu wypada BASIC XL.

WYKORZYSTANIE PAMIĘCI

Wszystkie interpretery oprócz Microsoft pozostawiają do dyspozycji użytkownika większy obszar pamięci niż Atari BASIC. Bezkonkurencyjny jest BASIC XE, który w trybie EXTEND udostępnia prawie 64 KB na program i 32 KB na dane. Nie ma jednak róży bez kolców. Turbo BASIC nie pozwala na założenie RAMdysku, a BASIC XE stawia alternatywę: albo RAMdysk, albo duży program. Dodatkowo BASIC XE nie działa na Atari 256XT po ustawieniu na 256 KB RAM. Natomiast nietypowe gospodarowanie pamięcią przez Turbo BASIC znacznie utrudnia korzystanie w programie z procedur maszynowych.

STRUKTURA PROGRAMU

Cechą decydującą o wygodzie użytkownika języka i pisania programów jest możliwość definiowania procedur wywoływanych przez nazwy zamiast numerów linii. Tylko BASIC XE i Turbo BASIC dają taką możliwość. BASIC XE pozwala jednocześnie na słosowanie zmiennych lokalnych, co znakomicie ułatwia pisanie programu. Natomiast w Turbo BASIC-u można także oznaczać etykietami linie dla GOTO, TRAP i RESTORE. Inną instrukcją strukturalną, zawartą w tych interpreterach jest WHILE. I tu Turbo BASIC ma dodatkowe możliwości: dysponuje on jeszcze pętlami REPEAT .. UNTIL i DO .. LOOP. Oprócz tego wszystkie interpretery umożliwiają stosowanie strukturalnej instrukcji IF: IF ... ELSE ... ENDIF.

INSTRUKCJE DYSKOWE

INTERPRETER	T1	T2	T3	T4	T5	T6	T7	T8	T9	DOKLADNOSC
Atari BASIC	2,3	7,3	19,7	24,1	27,7	40,2	60,9	423,2	1578,2	3.3806E-07
Turbo BASIC XL	0,8	2,8	7,6	8,5	9,5	14,8	24,8	56,5	347,1	3.3758E-07
Turbo BASIC XL (M)	0,8	2,3	7,1	8,1	8,5	13,8	23,9	56,0	344,0	3.3758E-07
BASIC XL	1,7	3,6	16,1	15,6	17,7	27,1	36,9	414,8	1555,2	3.3806E-07
BASIC XL (FAST)	1,6	2,9	15,5	14,9	15,6	23,1	32,5	414,0	1543,1	3.3806E-07
BASIC XE	1,4	4,1	16,2	15,9	19,1	27,7	40,4	417,7	1568,0	3.3806E-07
BASIC XE (FAST)	1,1	2,0	7,6	7,9	8,7	15,9	24,2	55,8	346,2	2.3804E-07
Microsoft BASIC	1,4	9,3	17,1	19,8	20,7	32,7	50,9	79,4	562,3	1.8824E-06

Dołączenie instrukcji dotyczących stacji dysków jest już właściwie standardem i wszystkie interpretery posiadają taki sam ich zestaw. Jedyne różnice polegają na zastosowaniu różnych słów kluczowych (zależnie od fantazji twórców interpretera). Na przykład do usuwania plików z dysku służą instrukcje: ERASE, DELETE lub KILL.

GRAFIKA

BASIC XL i XE posiadają specjalny zestaw instrukcji do obsługi grafiki graczy i pocisków (P/MG), co znakomicie ułatwia tworzenie gier. W programach użytkowych znacznie ważniejsze jest formatowanie wydruku (PRINT USING), a tej możliwości nie posiada tylko Turbo BASIC. Ma on natomiast jako jedyny bogaty zestaw instrukcji graficznych: FILLTO, PAINT, CIRCLE, TEXT i inne. W pozostałych interpreterach należy zamiast nich słosować własne procedury (do BASIC-a XE są dołączone odpowiednie procedury maszynowe na dysku lub kasecie).

FUNKCJE I PROCEDURY ARYTMETYCZNE

Procedury arytmetyczne Atari są bardzo powolne i stanowią jedną z największych wad Atari BASIC. Oprócz BASIC-a XL wszystkie pozostałe interpretery korzystają z własnych procedur arytmetycznych, które są od 3 do 5 razy szybsze. Zawierają one jednak bardzo podobny zestaw działań. Jedynie Microsoft BASIC został rozszerzony o funkcję TAN i możliwość definiowania funkcji przez użytkownika (DEF FN). Również w operacjach na tablicach tekstowych trudno wyróżnić jakiś interpreter (wszystkie dysponują podobnymi funkcjami). Wyraźnie inny jest tylko Microsoft BASIC, w którym możliwe jest deklarowanie tablic liczbowych i tekstowych wielowymiarowych. Poza tym BASIC XE posiada dwie bardzo interesujące instrukcje sortowania, które znacznie upraszczają programy użytkowe.

PODSUMOWANIE

Jak to zwykle bywa z językami programowania, każdy z przedstawionych interpreterów ma swoje wady i zalety, a jego użycie zależy od konkretnego zastosowania. Można jednak pokusić się o szacunkową ocenę. Microsoft BASIC mimo wielu niewątpliwych zalet posiada jednak równie dużo wad i jego zastosowanie jest ograniczone do niektórych programów obliczeniowych (np. wymagających wielowymiarowych tablic). BASIC XL jest jakby wcześniejszą wersją BASIC-a XE i polecałbym go tylko tym, którzy na pewno nie będą rozbudowywać systemu. Na placu boju pozostały więc jedynie BASIC XE i Turbo BASIC. Pierwszy z nich jest na cartridge'u, a drugi musi być każdorazowo wczytywany przed użyciem. Niestety ocena nie jest taka prosta. Otóż wiele instrukcji BASIC-a XE (w tym szybkie procedury arytmetyczne i wszystkie instrukcje P/MG) musi być wczytanych z dysku, a bez nich BASIC XE jest znacznie gorszy od Turbo. To jednak jeszcze nie wszystko: aby umożliwić korzystanie z BASIC-a XE użytkownikom magnetofonów (a jest ich w Polsce większość) w P.Z. Karen opracowano jego odmianę, która może odczytywać te dodatkowe instrukcje z kasety. Nie daje mu to jeszcze przewagi, gdyż oba interpretery wymagają wczytywania, a Turbo BASIC jest znacznie tańszy. Tak więc ostateczną decyzję każdy musi podjąć sam. Osobiście używam obu tych interpreterów na zmianę, a programy dla własnego użytku najczęściej piszę w ... Action!

```
100 REM *** TEST 1 ***
110 ? "START"
120 FOR K=1 TO 1000
130 NEXT K
140 ? "END";CHR$(253)
150 END
```

```
100 REM *** TEST 2 ***
110 ? "START"
120 K=0
130 K=K+1
140 IF K<1000 THEN 130
150 ? "END";CHR$(253)
160 END
```

```
100 REM *** TEST 3 ***
110 ? "START"
120 K=0
130 K=K+1
140 A=K/K*K+K-K
150 IF K<1000 THEN 130
160 ? "END";CHR$(253)
170 END
```

```
100 REM *** TEST 4 ***
110 ? "START"
120 K=0
130 K=K+1
140 A=K/2*3+4-5
150 IF K<1000 THEN 130
160 ? "END";CHR$(253)
170 END
```

```
100 REM *** TEST 5 ***
110 ? "START"
120 K=0
130 K=K+1
140 A=K/2*3+4-5
150 GOSUB 200
160 IF K<1000 THEN 130
170 ? "END";CHR$(253)
180 END
200 RETURN
```

```
100 REM *** TEST 6 ***
110 ? "START"
120 K=0
130 DIM M(5)
140 K=K+1
150 A=K/2*3+4-5
160 GOSUB 300
170 FOR L=1 TO 5
180 NEXT L
190 IF K<1000 THEN 140
200 ? "END";CHR$(253)
210 END
300 RETURN
```

```
100 REM *** TEST 7 ***
110 ? "START"
120 K=0
130 DIM M(5)
140 K=K+1
150 A=K/2*3+4-5
160 GOSUB 300
170 FOR L=1 TO 5
180 M(L)=A
190 NEXT L
200 IF K<1000 THEN 140
210 ? "END";CHR$(253)
220 END
300 RETURN
```

```
100 REM *** TEST 8 ***
110 ? "START"
120 K=0
130 K=K+1
140 A=K^2
150 B=LOG(K)
160 C=SIN(K)
170 IF K<1000 THEN 130
180 ? "END";CHR$(253)
190 END
```

```
100 REM *** TEST 9 ***
110 P=1.57
120 ? "START"
130 N=1000
140 D=0
150 A=0
160 S=P/N
170 FOR I=1 TO N
180 A=A+S
190 B=SIN(A)
200 GOSUB 500
210 B=COS(C)
220 GOSUB 520
230 GOSUB 550
240 C=ATN(B)
250 B=EXP(C)
260 C=LOG(B
270 [illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
```

Wojciech Zientara

KLAN AMSTRAD-SCHNEIDER

PRZEMIESZANE TRYBY

Poniższy program dodaje nowe rozkazy do BASIC-a pozwalając na pracę ekranu Amstrada w kilku trybach równocześnie.

Miłą cechą Amstradów jest możliwość podziału ekranu na cztery części (zwane dalej sekcjami) i nadawania im własnych trybów pracy. Kto widział grę SORCERY na Amstradzie, wie jakie efekty można uzyskać za pomocą tej melody W grze tej górna część ekranu pracuje w trybie 0. pozwalającym na uzyskanie 16-u kolorów, podczas gdy dolna część ekranu pracuje w trybie 1 i służy do wyświetlania tekstów

Mieszanie trybów osiągane jest za pomocą odpowiednich zmian wartości rejestru układu VGA (Video Gate Array) określającego tryb pracy ekranu.

Podział ekranu na sekcje jest stały. Wiersze 1-5 tworzą sekcję o numerze 0, wiersze 6-12 sekcję 1, wiersze 13-18 sekcję 2, a wiersze 19-25 sekcję 3.

Prezentowany program dodaje trzy nowe rozkazy do BASIC-A Po jego uruchomieniu program można wymazac z pamięci. Rozszerzenia będą aktywne oczywiście tylko do momentu wyłączenia komputera.

UWAGA! Dla bezpieczeństwa nie należy zapomnieć o zapisaniu wprowadzonego programu na taśmę lub dysk przed pierwszą próbą jego uruchomienia Wprowadzone rozkazy są rozkazami RSX rozpoczynającymi się znakiem: (SHIFT @)

.SETMO ,sekcja, tryb [,sekcja,tryb] — informuje system operacyjny Amstrada w jakim trybie ma wyświetlać daną sekcję Jednym rozkazem można nadać tryby jednej lub wielu sekcjom.

:SMODE ,tryb — służy do pisania po ekranie, informuje system w jakim trybie będziemy teraz wyprowadzac dane na ekran. Działanie tego rozkazu jest analogiczne jak rozkazu MODE z tym, że nie czyści ekranu. Używanie rozkazu MODE mija się z celem albowiem tryb ustawiony tym rozkazem zostanie zmieniany przy najbliższym przerwaniu

.NORMAL ,tryb — powrót do normalnego, całoekranowego trybu pracy Wskazane jest użycie tego rozkazu przed pierwszym użyciem rozkazu :SETMO.

Program utrzymując ekran w kilku trybach używa przerwań, które są blokowane podczas operacji na dysku lub taśmie Nie uzyskamy dobrych rezultatów próbując współpracować z pamięciami zewnętrznymi mając podzielony ekran.

Nie należy również dopuszczać do wysuwu ekranu (scroll), który zaburza synchronizację zmiany trybu pracy podczas przechodzenia pomiędzy sekcjami.

Załączony program demonstracyjny ułatwia zrozumienie sposobu użycia nowych rozkazów BASIC-a.

Na podstawie Popular Computing Weekly

Jarosław Borelowski

```
10 '     MIESZANIE TRYBOW PRACY CPC464
20 MEMORY 42300
30 FOR i=42301 TO 42521
40 READ a$:v=VAL("&"+a$)
50 cs=cs+v:POKE i,v
60 NEXT i
70 IF cs <> 23328 THEN PRINT "DATA ERROR - sprawdz listing";chr(7):STOP
80 CALL 42301:CLS:PRINT"OK ten program mozesz juz wykasowac -NEW":END
90 DATA 01,52,a5,21,2b,a6,cd,d1,bc,3e,00,32,2f,a6,2a,ec,bd,22,20,a6,c9,5d
100 DATA a5,c3,6e,a5,c3,dc,a5,c3,f7,a5,53,45,54,4d,cf,4e,4f,52,4d,41,cc,53
110 DATA 4d,4f,44,c5,00,cb,47,20,43,b7,28,40,f5,dd,7e,02,fe,06,30,2b,06,00
120 DATA 4f,21,1a,a6,09,dd,7e,00,77,3a,2f,a6,b7,20,1a,3e,ff,32,2f,a6,3e,00
130 DATA 32,19,a6,21,22,a6,06,81,0e,00,11,b6,a5,cd,19,bd,cd,e0,bc,f1,dd,23
140 DATA dd,23,dd,23,dd,23,3d,3d,20,c0,c9,3a,19,a6,3c,32,19,a6,47,fe,06,20
150 DATA 06,3e,00,32,19,a6,78,21,1a,a6,3d,06,00,4f,09,7e,d9,cb,89,cb,81,b1
160 DATA 4f,ed,49,d9,c9,21,22,a6,cd,e6,bc,3e,00,32,2f,a6,21,1a,a6,06,06,dd
170 DATA 7e,00,77,23,10,fc,cd,0e,bc,c9,21,0a,a6,22,ec,bd,dd,7e,00,cd,0e,bc
180 DATA 2a,20,a6,22,ec,bd,c9,cd,06,b9,cd,4f,0d,21,00,00,cd,3c,0b,c3,3c,0d
190 DATA 00
```

```
10 ' PROGRAM DEMONSTRACYJNY
20 ON ERROR GOTO 240
30 ON BREAK GOSUB 270
40 BORDER 0 : INK 0,0 : INK 1,13 : INK 2,26 : INK 3,6
50 :NORMAL,1  :                              'ustawienie trybu 1
60 CLS : :SETMO,0,2,1,0,2,0,3,1 :            'mieszanie trybow
70 :SMODE,2 : WINDOW 1,80,1,5
80 PEN 0 : PAPER 1 : CLS : PRINT:PRINT"  ";STRING$(76,42)
90 PRINT"  **   to jest przyklad mieszania trybow pracy CPC464 .     to jest MODE 2 **  "
100 PRINT "  " ; STRING$(76,"*")
110 :SMODE,1 : WINDOW 1,40,19,25
120 PEN 2 : PAPER 3 : CLS
130 PRINT : PRINT:PRINT"          MODE 1 - 40 kolumn w oknie"
140 LOCATE 17,5 : PEN 1:PRINT " BAJTEK" : PEN 1 : PAPER 0
150 :SMODE,0:ORIGIN 0,120:WINDOW 1,20,6,18
160 PEN 6 : LOCATE 2,13 : PRINT"press a key to cls "
170 DEG : FOR i=1 TO 500
180 IF INKEY$ <> "" THEN CLS : GOTO 160
190 col=INT(RND(1)*15+1)
200 x=INT(RND(1)*600) : y=INT(RND(1)*160) +20
210 FOR a=0 TO 360 STEP 30 : PLOT x,y,col:DRAW x+20*SIN(a),y+20*COS(a),col : NEXT a,i
220 :NORMAL ,1
230 END
240 IF ERR=28 THEN PRINT "trzeba zaladowac loader przed demo '!" : STOP
250 PRINT"error";ERR;"at line ";ERL
260 END
270 pen1:paper0::NORMAL,1:PRINT"* break *":END
```

PROGRAM KOJĄCY NERWY

Wszystkie redakcje pism poświęconych problematyce mikrokomputerowej zasypywane są listami czytelników piszących mniej więcej tak: „... przepisałem program drukowany w ostatnim numerze Waszego pisma i, niestety, mimo wielokrotnych prób i sprawdzania wydruku program nadal nie da się uruchomić. Chyba jest błąd w wydruku...".

Redakcje, współpracujące z odpowiedzialnymi autorami wiedzą, że nie jest możliwe, by błąd tkwił w wydruku sprawdzonego i uruchamianego wielokrotnie programu, lecz przekonanie zdesperowanego czytelnika o tym, że to on wprowadził błędy przy przepisywaniu, jest prawie niemożliwe Oczywiście można wymagać od autorów wprowadzania np. sumy kontrolnej i programów (podprogramów) obsługi błędów, jednak zabiegi takie prowadzą głównie do kontroli danych w liniach DATA i to dopiero poczas uruchamiania programów. Proponowany dzisiaj WERYFIKATOR jest programem stosowanym przez francuskie pismo Amstrad Magazine i zachodnioniemieckie CPC Schneider International, umożliwiającym kontrolę wprowadzanych linii programu „na bieżąco", niezależnie od tego czy są to linie DATA, czy linie zawierające słowa kluczowe lub komentarze. Każdy znak w linii „sumowany" jest z następnym i w momencie wprowadzania linii do pamięci (klawisz RETURN lub ENTER) generowana jest suma konrolna ujęta w nawiasy kwadratowe. Przy przyjętym przez te pisma drukowaniem programów łącznie z sumami kontrolnymi podawanymi dla każdej linii możliwa jest natychmiastowa kontrola poprawności wprowadzania kolejnych linii przez porównanie sumy kontrolnej wydruku i sumy kontrolnej linii wpisywanej przez użytkownika."

Warto więc po raz ostatni skupić się i przepisać DOKŁADNIE podany program, ponieważ przy następnych wydrukach będę już podawał sumy kontrolne Prawidłowo wpisany program po uruchomieniu oferuje cztery rozkazy: | ON /uaktywniony od razu w linii 180, | OFF — wyłączający weryfikator, | CHECK,2 — odpowiadający komendzie LIST ale z wyświetlaniem sumy kontrolnej i | CHECK,8 pozwalający na uzyskanie wydruku linii programu na drukarce z podawaniem sumy kontrolnej. Po uruchomieniu weryfikatora można przystąpić do przepisywania interesujących nas programów pod warunkiem, że ich wydruk został wykonany również przy użyciu WERYFIKATORA. Posiadacze CPC 464 i 664 powinni zmienić niektóre linie DATA zgodnie z dołączonym wykazem. Niecierpliwi mogą sprawdzić działanie programu wpisując dowolne linie lub rozkazy w trybie bezpośrednim. Po każdym wciśnięciu klawisza RETURN lub ENTER powinna się pojawić liczba w nawiasach kwadratowych. Cierpliwych zapraszam za miesiąc.

Na podstawie Amstrad/Schneider 1/87

Wojciech Ziółek

```
[illegible] Weryfikator V.2 dla CPC 6128
[illegible] MEMORY [illegible]
[illegible] FOR [illegible] TO &A607
[illegible] READ [illegible]
140 POKE a%,VAL("&"+b$)
[illegible]
160 CLS:PRINT
[illegible] PRINT "Weryfikator V.2 zainstalowany"
[illegible] PRINT "Zaladuj swoj program"
[illegible] PRINT "Rozkaz :CHECK,2 listuje program"
[illegible] PRINT "i sprawdza sume kontrolna."
178 PRINT "Rozkaz :CHECK,8 listuje program"
[illegible] PRINT "z suma i drukuje na drukarce."
180 CALL &A5[illegible]:ON
190 DELETE
200 DATA 21,09,a5,[illegible],0d,a5,c3,d1
210 DATA bc,[illegible],00,00,00,18,45,c3
220 DATA 2a,a5,c3,2f,a5,c[illegible],43,a5
230 DATA 41,c[illegible],4f,46,c6,43,48,45
240 DATA 43,cb,[illegible],00,cf,02,ac,c3
250 DATA a8,a5,[illegible],27,a5,18,03,21
260 DATA 24,a5,28,06,cd,00,b9,c[illegible]
270 DATA 49,cb,11,5e,bd,01,03,0[illegible]
280 DATA ed,b0,c9,4f,cd,00,b9,0d
290 DATA 28,08,[illegible],[illegible],[illegible],dd,7e,02
300 DATA 18,04,7b,11,01,00,cd,a6
310 DATA [illegible],cd,64,[illegible],eb,4e,23,46
320 DATA [illegible],5e,23,56,e1,78,b1,c8
330 DATA cd,[illegible],c4,a5,09,e3,cd,54
340 DATA e[illegible],[illegible],8a,ac,cd,7a,a5,e1
350 DATA 18,[illegible],e5,cd,ba,a5,e5,cd
360 DATA 98,a5,cd,5[illegible],f5,e[illegible],cd,f6
370 DATA a5,[illegible]d,98,c[illegible],e1,7e,a7,c8
380 DATA cd,98,a5,cd,98,c3,18,f5
390 DATA 3a,09,ac,d6,08,47,7e,a7
400 DATA c8,cd,1d,e7,23,10,f7,c9
410 DATA cd,[illegible]4,a5,f5,c5,d5,e5,cd
420 DATA ba,a5,cd,f6,a5,e1,d1,c1
430 DATA f1,[illegible]9,[illegible],1b,af,47,[illegible],6f
440 DATA 2[illegible],[illegible],[illegible],a5,1[illegible],1a,d6,30
450 DATA 38,04,fe,0a,[illegible],f6,1a,1[illegible]
460 DATA a7,c8,4f,3a,23,a5,a1,fe
470 DATA 20,[illegible]8,[illegible]3,79,fe,22,20,07
480 DATA [illegible]a,23,a5,2f,32,2[illegible],a5,3a
490 DATA 23,a5,a7,79,c4,[illegible],ff,4f
500 DATA ad,07,6f,09,18,d8,3e,20
510 DATA cd,a0,c3,3e,5b,cd,a0,c3
520 DATA cd,44,ef,3e,5d,c3,a0,c3
```

```
[illegible] linii DATA dla CPC 464
240 DATA 43,cb,00,00,cf,[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
4[illegible] DATA [illegible]
450 DATA [illegible]
510 DATA [illegible]
[illegible]
```

```
[illegible] DATA dla CPC 664
[illegible]
```

BITIMAGE

(GRAFICZNY ZRZUT ZBIORU)

Badanie cudzych zbiorów lub programów, zwłaszcza w kodzie maszynowym, jest zajęciem żmudnym, ale niezwykle pouczającym. Oferowane firmowo do tego celu programy typu debbuger (np. SIO w systemie CP/M) pozwalają otrzymać zrzut znakowy zbioru lub przetłumaczony na mnemoniki assemblera kod wynikowy.

Prezentowany w niniejszym artykule program Bitimage daje nową możliwość — graficzny zrzut zbioru przy użyciu drukarki, pracującej w specjalnym trybie. Zaletą tego sposobu jest niezwykła zwartość wydruków. Obraz programu zajmującego 30 kB mieści się na jednej stronie formatu A4. Każdy bajt zbioru drukowany jest igiełkami głowicy w wąskiej kolumnie. Zerowy bit nie zostawia śladu, a bit ustawiony na wartość 1 jest małą kropką. Bez problemów w jednym wierszu można zmieścić obraz ponad 700 bajtów. Nie namawiam nikogo do desassemblacji tak otrzymanego wydruku, ale może on być bardzo użyteczny, gdy chcemy mieć ogólne pojęcie o badanym zbiorze. Łatwo odpowiedzieć na pytanie gdzie są puste miejsca, lub gdzie jest generator znaków ekranu. Można także porównywać zbiory ze sobą znacznie skuteczniej niż przy pomocy zwykłego programu COMPARE, który przy napotkaniu pierwszej niezgodności daje enigmatyczną odpowiedź: „BAD COMPARE, FILES DO NOT MATCH", co w tłumaczeniu na język polski oznacza: złe porównanie, zbiory nie pasują do siebie.

Program został napisany w TURBO-PASCAL'u i przetestowany na komputerze AMSTRAD PCW 8256. Oczywiście osoby posługujące się innym językiem programowania mogą sobie przetłumaczyć program, ponieważ realizowany przez niego algorytm jest bardzo prosty. Dwa istotne problemy to:

1. zastosowanie trybu graficznego drukarki, przy pomocy instrukcji: write (lst, chr (27), 'K', chr(n0), chr(n1), chr(b1), chr(b2),... ...);
   gdzie
   n0 + 256 • n1 — ilość bajtów do wydrukowania
   b1,b2,..... — kolejne bajty do druku
2. czytanie zbioru z dysku, realizowane w PASCAL'u instrukcją BLOCKREAD.

Ilustracją zastosowanie programu jest obraz fragmentu zbioru systemowego J14CPM3.EMS zawierający generator znaków ekranu komputera AMSTRAD PCW 8256.

Janusz Mayer

```
program BitImage;
(*****************************************************************)
(*   program do zrzutu graficznego zawartości zbioru dyskowego *)
(*                          (C) JM                              *)
(*                        listopad 1987                         *)
(*****************************************************************)
const
     Bufsize = 64;     Recsize = 128;
type
     HexStr  = string(.4.);
var
     Source                : file;
     Buffer                : array(.1..BufSize,1..Recsize.) of byte;
     RecsRead, RecsToProc,
             address       : integer;
     SourceName            : String(.20.);

function Hex(Number : integer; Bytes : integer) : HexStr;
const
     T   : array(.0..15.) of char = '0123456789ABCDEF';
var
     D   : integer;
     H   : HexStr;
begin
      H(.0.) := chr(Bytes+Bytes);
      for d := Bytes + Bytes downto 1
      do begin
             H(.d.) := T(.Number and 15.);
             Number := Number shr 4;
         end;
      Hex := H;
end; (* of Hex *)

procedure ProcessBuffer;
var
     r,i, q  : integer;
begin
     write(lst,chr(27),'0');        (* 8 linii/cal *)
     q:=0;
     for r := 1 to RecsRead
     do begin
             if q=0 then write(lst,hex(address,2):5);
             write(lst,chr(27),'K',chr($80),chr(0));   (* tryb graficzny drukarki *)
             for i:=1 to $80
             do write(lst,chr(buffer(.r,i.)));
             q := q + 1;
             address := address + $80;
             if q=3
             then begin
                    q := 0; writeln(lst);
                  end;
        end;
end; (* of ProcessBuffer *)

begin (* main *)
      writeln(' Zrzut graficzny zbioru dyskowego ');
      write('nazwa zbioru: '); readln(SourceName);
      writeln(lst,SourceName:40);
      assign(Source,SourceName);
      reset(Source);
      RecsToProc := FileSize(Source);
      writeln('Record(y): ',RecsToProc,'  Byte''y: ',128.0*RecsToProc:4:0);
      reset(Source);
      address := $100;
      While RecsToProc>0
      do begin
             BlockRead(Source,Buffer,BufSize,RecsRead);
             RecsToProc := RecsToProc - RecsRead;
             ProcessBuffer;
             writeln(lst);
         end;
end.
```

```
2100
2280
2400
2580
2700
2880
2A00
2B80
2D00
2E80
3000
3180
3300
3480
3600
3780
3900
3A80
3C00
3D80
3F00
```

EQUINOX

Na dalekiej planecie Equinox, gdzie dzień jest zawsze równy nocy, zbudowano kopalnię radioaktywnego uranu. Kopalnia została w pełni zautomatyzowana, tak, że w procesie wydobycia nie brał udziału żaden człowiek.

Każdy z ośmiu sektorów kopalni wydobywa i magazynuje uran niezależnie od innych. Po zakończeniu wydobycia pojemniki z tym pierwiastkiem będą przekazane z każdego sektora do magazynu. Niestety, teren kopalni stał się miejscem zamieszkania Equinitów — złośliwych mieszkańców planety. Unieruchomili oni kopalnię w ważnym momencie procesu wydobywczego — w chwili, gdy uran miał być przekazywany z sektorów do magazynu. I tu rozpoczyna się misja bojowo-zadaniowego robota o nazwie PAUL, maszyny tej samej serii, co sławny MAUL, członek Enigma Force. Zadaniem Paula jest przesłanie wszystkich ośmiu pojemników z uranem do magazynu za pośrednictwem windy oznaczonej na mapie kolorem czerwonym. Ma on ściśle określony czas na przetransportowanie każdego z pojemników. Jeżeli nie zdąży, promieniowanie uranu unicestwi go. Tym samym wszelkie nadzieje na uratowanie planety zginą razem z nim.

Po przesłaniu pojemnika do magazynu zawarty w nim uran nie będzie miał szkodliwego działania, ponieważ magazyn jest ekranowany grubą warstwą ołowiu.

Na swej drodze Paul napotka głębokie szyby z magnesem na górze. Wejście do szybu powoduje wciągnięcie robota.

Dotarcie do pewnych części kopalni możliwe jest tylko poprzez teleportację. Trzeba jednak za nią płacić — jedna moneta za transport w tę i z powrotem. Może się zdarzyć, że Paulowi zabraknie monet. Wtedy jedynym wyjściem jest... przetopienie się w tyglu na pieniążek. Odbiera to oczywiście robotowi jedną zapasową powłokę. Paul ma tylko dwie zapasowe powłoki więc powinien bardzo uważać.

Niektóre przejścia są zasypane. W takiej sytuacji ratunkiem jest łopata lub dynamit. Te z kolei znaleźć można w zamkniętych szafach pancernych, które dają się otworzyć jedynie wiertarką. Czasami można znaleźć klucz do drzwi. Kłopoty pojawiają się na przykład przy jednej łopacie i dwóch zatorach. Wybór nie jest łatwy — pomyłka właściwie kończy grę.

Każde zetknięcie się z Equinitem zmniejsza energię Paula. Ratunkiem jest znaleziony akumulator regenerujący utraconą siłę. Pomocne będą też beczki z paliwem, ponieważ silnik pozwalający robotowi unosić się zużywa go dużo. Znalezione karty z numerami od 1 do 8 umożliwiają poruszanie się między sektorami poprzez centralę.

Paul dysponuje tylko jedną kleszcznią, więc musi trafnie dobierać, co ma nieść.

Grając w Equinox używać można klawiatury lub joysticka. Wciśnięcie GÓRA powoduje unoszenie się robota a DÓŁ — podniesienie przedmiotu lub użycie tegoż. Dam Ci jeszcze jedną radę: będąc w pierwszym sektorze znajdź kartę PETE, wróć do pomieszczenia startu i będąc w lewym górnym rogu wciśnij jednocześnie klawisze R, N, C. Staniesz się nieśmiertelny lecz czas będzie płynął.

Firma: Mikro-Gen

Komputer: Spectrum 48/+, Commodore 64, Amstrad/Schneider

Marcin Przasnyski



1

start

2

4

taśmociąg

drzwi

klucz do dr

zapora laser



bezpiecznik

kanister z uranem - odnieść do

dynamit-wysadzać

beczka z paliwem

akumulator

moneta do teleportacji

miejsce na bezpiecznik

łopata-kopać

wiertarka-wiercić

zator poziomy

zator pionowy

przejście-zapadnia

teleportacja

przejście między sektorami

winda

tygiel

winda do mag. kanistrów

szafa

karta nieśmiertelności

karta wstępu na sektor

10 BAJTKOWA LISTA PRZEBOJÓW (1/88)

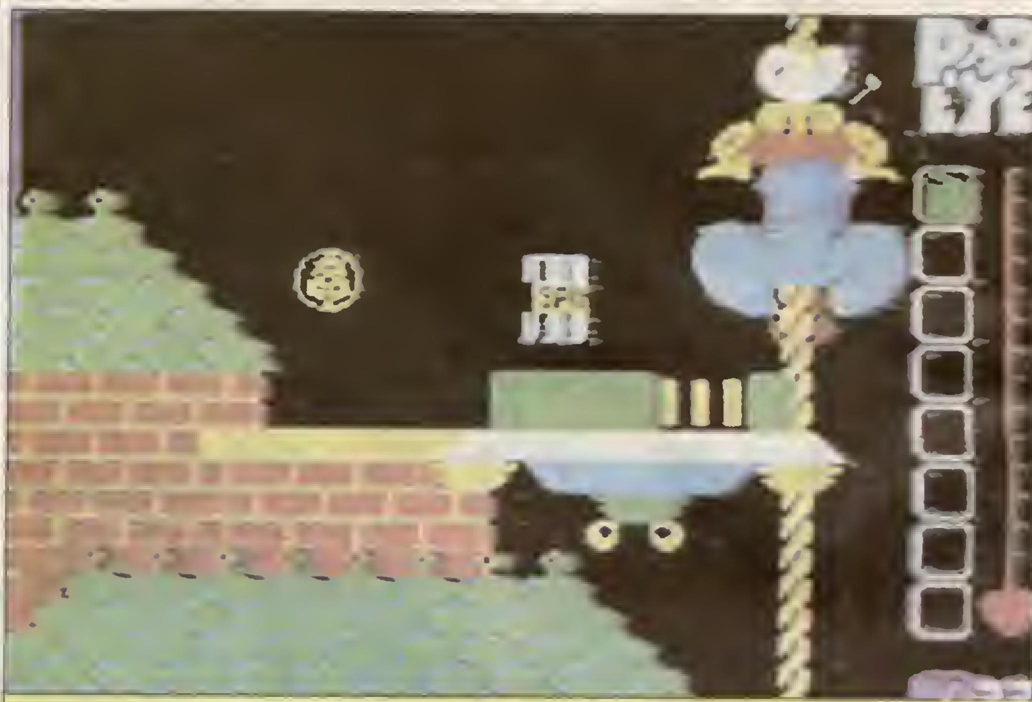
Boulder Dash znowu na pierwszym miejscu! Powrót na czoło listy zapewniło mu chyba zdobycie pierwszego miejsca w Złotej Dziesiątce roku 1987. I chociaż gra jest dosyć stara, to każdy wraca do niej z przyjemnością. Na pierwsze tegoroczne notowanie otrzymaliśmy łącznie 3711 propozycji na 115 tytułów.

			ATARI	AMSTRAD	COMMODORE	SPECTRUM
1	BOULDER DASH	!	x	x	x	x
2	ANTIRIAD	!		x	x	x
3	MISS PACMAN	↓	x	x	x	x
4	BARBARIAN		x	x	x	x
5	NIGHT SHADE	!		x	x	x
6	POP EYE	!		x	x	x
7	GLADIATOR	↑		x	x	x
8	GREAT ESCAPE	↓		x	x	x
9	WIZARD'S LAIR	↓		x	x	x
10	REVOLUTION	↓		x	x	x

Nagrody wylosowali: Monika Rembikowska i Piotr Jodłowski



Kopertę nadesłał Tomasz Mikus z Płocka



POP EYE

Właśnie wróciłem z długiego rejsu dookoła świata i raźno pobiegłem do domu, lecz zastałam go zamkniętym na cztery spusty. Żona była wewnątrz, bo pod wycieraczką nie znalazłem klucza. Zrezygnowany usiadłem na schodkach w milczeniu ssąc wygasłą fajeczkę. Nagle okno za mną otworzyło się z hukiem i usłyszałem gderliwy głos żony. „Gdzie się włóczyłeś tak długo leniu! Ja cię nauczę zajmować się domem a nie wozić brzuch po zamorskich knajpach!". Chciałem podbiec do okna, przeprosić, lecz to zatrzasnęło się

Co robić? Żona mnie odtrąciła, na morze szybko nie wrócę, gdzie się więc podzieję? Schronię się w latarni morskiej, potem pomyślę, co robić dalej.

Poszedłem nad morze. Na latarni dostrzegłem puszkę szpinaku — mej ulubionej potrawy i dwa pierniki w kształcie serduszek. Gdy je połknąłem, poczułem, jak otucha wypełnia moje serce. Pobiegłem do domu spróbować jeszcze raz przeprosić żonę. Ujrzałem ją w oknie... radośnie uśmiechniętą. Po moim pierwszym pocałunku jej uśmiech znikł, po drugim okno zamknęło się i poczułem na policzku chłód dobrze wyheblowanych desek.

Wiedziałem już, co muszę zrobić. Mam zebrać wszystkie pierniki (jest ich 26) i przeprosić żonę całując ją 26 razy.

Skądś nadleciał ogromny sęp i zanim zdążyłem się zorientować, powalił mnie na ziemię. Wyciągnąłem z kieszeni puszkę szpinaku i szybko wsypałem sobie do ust porcję ożywczej potrawy. To postawiło mnie na nogi.

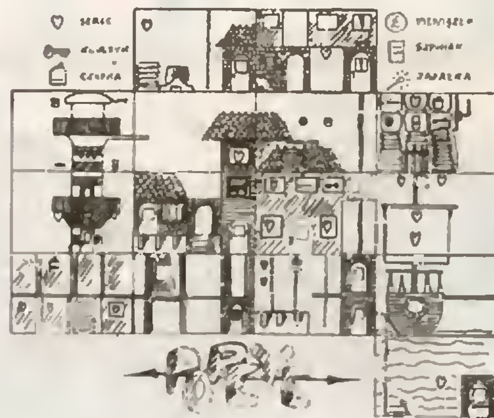
Za sępem kroczył gruby bosman, skądinąd znany mi handlarz niewolnikami. Zdążyłem ukryć się za załomem muru tak, że mnie nie zauważył. Od tej chwili wszystko układało się pomyślnie. Klucze pasowały do niektórych zamkniętych drzwi, czapka-niewidka uchroniła mnie przed ogniem z paszczy smoka — lokatora strychu, zapałka uruchomiła armatę, która swym zielonym płomieniem wstrzeliła mnie na górę. Nie udało mi się tylko schować przed czarownicą — siostrą bosmana, ale od czego szpinak?

Poważny kłopot miałem z „Jednorękim Bandytą" — automatem do gry, który widziałem już niejednokrotnie za granicą. Po wrzuceniu pieniążka i pociągnięciu za dźwignię w okienkach pojawiać się zaczęły różne przedmioty, także literki, głównie P, O, E i Y. W końcu odkryłem, że muszę ułożyć napis POP EYE — moje przezwisko, którego bardzo nie lubiłem. Zauważyłem też, że gdy stanę przed „Bandytą" tak, by moja głowa była na wysokości pewnego okienka i potem uruchomię automat, przedmiot w tym okienku nie będzie się zmieniał. Tak ułożyłem cały napis i dostałem w nagrodę sześć pierników.

Gdy po zebraniu wszystkich serduszek pobiegłem do okna i po raz ostatni pocałowałem żonę, drzwi otworzyły się, zapraszając do środka. Oboje już w zgodzie zasiedliśmy do powitalnej uczty i odtąd żyliśmy długo i szczęśliwie.

Firma: Macmillan Software
Komputer: ZX Spectrum 48/+, Amstrad/Schneider

(m.p.)



KRÓL I KRÓLOWA GIER



Monika Rembikowska lat 13, uczennica VI klasy Szkoły Podstawowej nr 260 w Warszawie, zam. Warszawa, ul. Królowej Marysieńki 21 m 51. Ma komputer Commodore 64, jej zainteresowania to języki — angielski i esperanto. Koresponduje w tych językach z koleżankami i kolegami z zagranicy.



Piotr Jodłowski lat 13, uczeń Szkoły Podst. nr 27 w Krakowie. Chodzi do VII klasy, ma komputer ATARI i wspólnie z kolegami opracowuje programy gier i edukacyjne. Niemniej jednak jego największym hobby jest koszykówka.

DAMBUSTERS

Już od początku roku 1943 alianci wspólnie odbywali wielkie dywanowe naloty na ważne strategicznie punkty Rzeszy.

W marcu 1943 roku powstał projekt zniszczenia elektrowni i zapór wodnych dostarczających energii fabrykom zbrojeniowym w środkowym biegu Renu. Przygotowywania do akcji trwały szereg miesięcy i w tym czasie opracowano specjalny model bomby, która odpowiednio rzucona, ślizgając się po wodzie zniszczy grodzie zapór.

Jest 16 maja 1943 godz. 21.15, grupa ciężkich bombowców typu Avro Lancaster wystartowała z bojowego lotniska w Scompton i skierowała się na pod.-wsch. przez Francję do Niemiec. Po wielu treningach i innych przygotowaniach brytyjski 617 Dywizjon Bombowy rozpoczął akcję która bezpośrednio miała przybliżyć koniec wojny i klęskę Rzeszy.

Zniszczenie elektrowni wodnych w Zagłębiu Ruhry oprócz odcięcia źródeł energii spowodowałoby zalanie i zniszczenie ok. 40% militarnego potencjału Niemiec.

A jaka jest Twoja rola w tej akcji?

Jesteś jednocześnie: pilotem, nawigatorem, przednim i tylnym strzelcem, bombardierem i mechanikiem pokładowym w swojej maszynie. Miejsce w którym aktualnie jesteś w samolocie oznacza mały wyświetlacz na dole ekranu a przenieść się do innego pomieszczenia można za pomocą siedmiu klawiszy od „Q" do „U".

Przed tobą trudny i wyczerpujący lot przez p. Europę.

Sama gra po wybraniu sposobu kontroli nad maszyną rozpoczyna się od wyboru: wskaźników — cyfrowych (DIGITAL) lub normalnych wychyłowych, i rodzaju lotu:

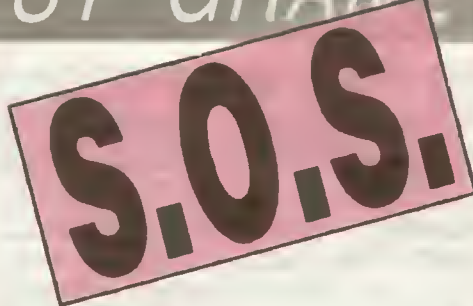
TAKE OFF — lot ze startem
FLIGHT — maszyna jest już w powietrzu gotowa do jej przejęcia
PRACTICE — lot treningowy (jak wszystkie przed miesiącem)

Jak będzie przebiegał Twój lot?

1. Startujesz z lotniska Scompton (12 mil od wschodniego wybrzeża Anglii) i po linii prostej ku brzegom Holandii. Kurs po jakim będziesz leciał stawiasz kursorem, który umieszczasz na mapie w miejscu przeznaczenia. Wtedy pilot automatycznie odnajduje właściwy kierunek na kompasie pokładowym. Pamiętaj jednak aby nie wybrać drogi po linii prostej i omijać stanowiska artylerii P-Lot i obszary strzeżone z powietrza.
2. Start — to nic trudnego ale pamiętaj że o twoim położeniu względem Ziemi dowiesz się tylko z przyrządów.
3. Jeżeli zostaniesz wykryty przez reflektory wroga a twój strzelec pokładowy nie zrobi z nimi porządku; będziesz musiał przeprowadzić serię kontrolowanych uników. Jeżeli jednak światło reflektora będzie nie do zgubienia to stajesz się automatycznie celem całej naziemnej artylerii.
4. Unikaj też lotnisk lub większych skupisk miejskich bo gdy jakiś Me-110 siądzie Ci na ogonie to nie będzie łatwo się go pozbyć. Strzelaj celnie i starannie odmierzaj poprawki.
5. Inną niemiłą niespodzianką jest balon zaporowy. Choć podobno w czasie pierwszej wojny światowej byli piloci potrafiący wylądować na takim balonie swoją maszyną to Tobie nie radzę tego próbować. Strzelaj do balonów z broni pokładowej.
6. Pierwszy mechanik pracuje praktycznie samodzielnie do momentu zrzutu. Na czas lotu ustaw obroty silników na 9600 (gdy nastawisz ponad 10000, po prostu je spalisz) co daje średnią prędkość przelotu 230 mph.
   Wskaźniki:
   Pierwszy rząd u góry — doładowanie silników.
   Drugi rząd od góry — obroty silników
   Przełączniki w prawym górnym rogu — wyłączniki silników.
   Pierwsze cztery suwaki od lewej — sterowniki.
   Następne cztery — sterowniki doładowania.
   Ustawianie przez umieszczenie kursora przy odpowiednim wskaźniku i ruch góra-dół aż do uzyskania odpowiedniego położenia.
7. Drugi mechanik kontroluje zapasy paliwa i ich zużycie, podnosi lub opuszcza klapy. Trzyma pieczę nad sterownikiem korekcyjnym, który może być użyty gdy wysiądzie któryś z silników.
   Wskaźniki:
   Rząd u góry — stan paliwa.
   Okrągły na środku tablicy — położenie klap.
   Okrągły przełącznik — klapy.
   Przełącznik kwadratowy — podwozie.
8. Mapa w kabinie nawigatora. Obranie kursu musi być bardzo dokładne gdyż nawet mały błąd może być już tym ostatnim.
9. Nad celem musisz spokojnie krążyć dopóki strzałka kompasu nie zrówna się z linią znacznika kierunku. Ustaw wysokość na 51 stóp. Zredukuj trochę obroty silnika i sprawdź pracę pierwszego mechanika.
10. Teraz czas na robotę dla celowniczego. Na początek włącz silnik transportujący (przełącznik z prawej) w swojej bombie. Uważnie obserwuj światła celownika i gdy obie plamki pokryją się oznaczać to będzie że jesteś na dobrej wysokości po obliczeniu odległości do tamy możesz przejść do pkt 11.
11. Jest już tama. Gdy na całej długości pokryje się ona z linią znacznika odległości na tablicy przedniego strzelca i gdy obie wieżyczki tamy znajdą się w obrębie tych znaczników możesz uwolnić bombę.
12. Gdy nie trafisz w tamę możesz już tylko rozpocząć grę od początku gdyż drugie podejście jest niemożliwe.
    Gdy trafisz, tama wyleci w powietrze woda ruszy i otrzymasz oficjalne gratulacje oraz końcowy wynik w punktach.
    Polecam przed każdym lotem spróbować treningu aby akcja była pewna na 99%, a ten 1% może zależeć już od pogody. Powodzenia.

Producent: US Gold
Komputer: Amstrad/Schneider, Commodore 64/128, ZX-Spectrum 48/+/128/+2/+3.

M.1



Mam komputer Atari. Nie mogę przejść komnaty Nomen Luni w grze JET SET WILLY i proszę o nieśmiertelność do niej. Pomóżcie!

Marek Lach
os. 700-lecia 1/22
34-300 Żywiec

Od kilku tygodni poszukuję nieśmiertelności i innych ułatwień w grach. DYNAMITE DAN, BRUCE LEE, MINER 2049, PARK PATROL i MIKIE na C64.

Piotr Zastróżny
ul. Langiewicza 7
81-737 Sopot — Wyścigi

Kto pomoże mi dokończyć grę PRICE OF MAGIC na Atari? Nie mogę przejść przez ciemność za magicznymi drzwiami.

Arkadiusz Stawarz
ul. Macedońska 37/6
51-113 Wrocław

Liczę na Waszą pomoc w uzyskaniu dokładnego opisu gier GHOSTBUSTERS i MONTEZUMAS na Atari. Jak zdobyć nieśmiertelność do tych gier?

Piotr Winiarski
ul. Idzikowskiego 3
37-700 Przemyśl

Ratunku! Nie wiem, jak pokonać takie gry na Atari jak: EASTERN FRONT (gdzie doprowadzić niemieckie dywizje?), MOLECULE MAN (jak znaleźć pieniądze lub bombę? Do czego używać koszyczka?), HACKER, GHOSTBUSTERS

Grzegorz Nelepa
ul. Pszczyńska 112c/29
44-100 Gliwice

Oprócz gier interesują mnie też programy. Kiedyś zetknąłem się z grą POGO JOE i nie mogę jej uruchomić. W zamian dam opisy do gier ROAD RACE, NINJA, BRUCE LEE

Mateusz Kasprzak
ul. 1-go Maja 67/17
32-100 Proszowice

Jak rozpocząć gry NIGHT SHADE i DOWN OF CIVILIZATION i jak w nie grać? Szukam też mapy do gry CHIMERA. Co zrobić w SPY Vs SPY po zebraniu i złożeniu rakiety?

Wojciech Musiał
ul. Jagiełły 28
77-430 Krajewka
woj. pilskie

Jestem posiadaczem ATARI 65XE i bardzo proszę o udzielenie mi kilku wskazówek o grze MONTEZUMAS REVENGE i uzyskiwanie nieśmiertelności w tej grze. Służę opisami gier. CAVE LOOR i FORT APOCALYPSE

Krzysztof Piskorczyk
ul. Bielska 41 m 19
43-400 Cieszyn

Mam wielkie trudności z grą TOMAHAWK i dlatego szukam dokładnego opisu. W zamian prześlę opisy do gier: JETMAN, PYJAMARAMA, PSSST, BRUCE LEE

Adam Muzyka
ul. Piastowska 45/8
50-361 Wrocław

Mam komputer TK90X. Jest on kompatybilny ze Spectrum. Proszę o opis gry PYJAMARAMA. W zamian proponuję opisy gier: SCOOBY DOO, XEVIOUS, REVOLUTION i innych. Chciałbym korespondować z użytkownikami komputerów MSX. Posiadam wiele programów w MSX Basic.

Michał Pstrucha
ul. Obr. Stalingradu 116/51
40-613 Katowice 8

Proszę o pomoc w grach. SPY Vs SPY II, RAMBO, ZORRO, na C64. Nie wiem, jak w RAMBO zlikwidować nieśmiertelność. Liczę na waszą pomoc

Zofia Rupala
ul. Orkana 9/50
25-548 Kielce

LISTY DO LISTY

Z zainteresowaniem śledzę rubrykę CO JEST GRANE. Od niedawna posiadam Spectrum i jestem zafascynowany CHUCKIE EGG. Opis tej gry zamieszczony został w 10 „Bajtku". Jest to z pewnością wersja na Amstrada, a ja chciałbym podzielić się moimi uwagami:

1. **W moim programie brak programu demonstracyjnego, dlatego też nie kwestionuję czy jest naprawdę (w co wątpił pan „tp") 30 pomieszczeń, ale wierz w to, bo sam straciłem „ostatnie życie" w 21.**
2. **Chcę wyjaśnić budowę kurnika. Otóż: Najpierw trzeba przejść 8 pomieszczeń, w których występują tylko kury. Natykając się na nie można stracić życie. Można również ją stracić, gdy nieopacznie wskoczymy do klatki z olbrzymią kaczką.**
   Kaczka ta pojawia się na 9 poziomie (którego platformy i drabinki rozłożone jak na 1 poziomie).
   Kaczka ta lata ciągle za nami, ale można (co uczyniłem) ją przechytrzyć. W poziomach od 9-16 „występuje" tylko kaczka. Kury znikają. Wszystkie poziomy powtarzają się jak widać co 8. Tak więc poziom 17 to ten pierwszy z ale bo chodzą jak dawniej kury i... jak dawniej lata ciągle w naszym kierunku kaczka.

CHUCKIE EGG jest grą pasjonującą. Nie posiadając żadnych POKE-ów doszedłem do 21 poziomu, zdobywając 223 600 punktów.

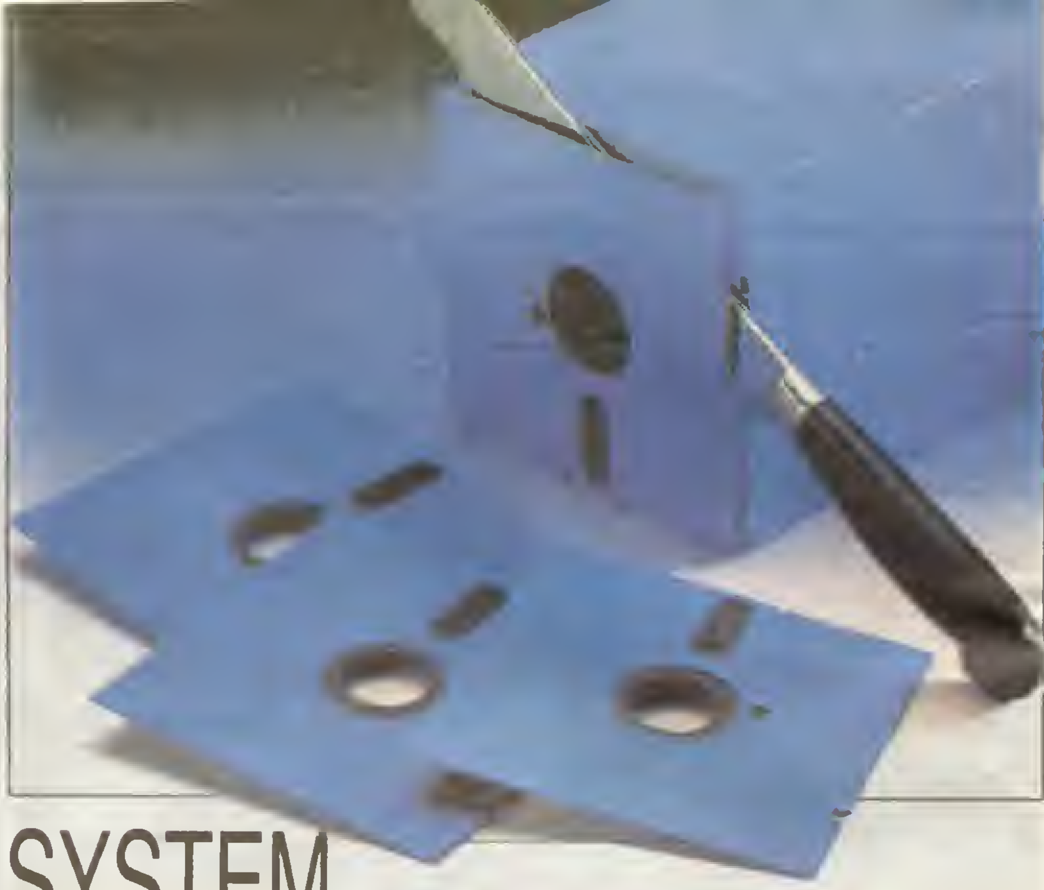
Po przejściu 5 poziomu bez utraty życia (lub innych kolejnych) często możemy otrzymać premię w postaci nowego życia — kapelusika.

A skoro mówimy o premiach to BONUS od 1 do 9 rośnie o tysiąc — na 9 platformie BONUS wynosi 9000 i zmniejsza się z upływem czasu.

Arkadiusz Szyszko
ul. Dworcowa 15 m 3
14-400 Pasłęk

NASTĘPNY KROK

Miesiąc temu rozpoczęliśmy wielką wędrówkę w czasie, aby przyjrzeć się powstaniu i rozwojowi systemów operacyjnych. Doszliśmy do etapu, gdy program zarządzający pracą maszyny stał się niezbędny. Jego zadanie to przyjmowanie od użytkownika poleceń wydawanych w jakimś, ogólnie mówiąc, cywilizowanym języku i ich wykonywanie. Chodzi oczywiście o polecenia sterujące pracą maszyny, a więc np. załaduj program do pamięci, wykonaj program, prześlij dane na drukarkę, itd.



SYSTEM OPERACYJNY (2)

Zauważmy, że takie właśnie funkcje są dostępne na prostych mikrokomputerach, choć na ogół nie w formie wyodrębnionego programu, lecz dołączone do interpretera BASIC-a. Widać tu również sygnalizowany już fakt wydzielenia na system części zasobów — pamięć ROM, zawierająca interpreter a wraz z nim system, zajmuje część przestrzeni adresowej procesora, a więc pomniejsza obszar pamięci maszyny dostępnej dla programów.

Wracamy do historii. Taśma magnetyczna to funkcjonalny odpowiednik perforowanej taśmy papierowej. Jest oczywiście kilka różnic — magnetyczna jest szybsza, pojemniejsza, dużo pewniejsza w działaniu, mniej podatna na zniszczenie, nawet oszczędniejsza — można ją wykorzystywać, tzn. nagrywać i kasować, wielokrotnie. Co więc decyduje o tym, że wrzucamy obie do wspólnego worka? Jedna zasadnicza cecha: w obu przypadkach dostęp do zapisanych informacji jest sekwencyjny. To znaczy, że aby przeczytać coś co jest zapisane na końcu, muszę najpierw przejść przez wszystko, co było zapisane uprzednio. Natomiast nowe informacje muszę dopisywać na końcu — nie wolno tego zrobić w środku istniejącego zapisu. Oczywiście spowalnia to bardzo dostęp do danych. Może nie dla wszystkich zadań — są zadania doskonale pasujące do pamięci sekwencyjnej. Jednak dla bardzo wielu ważnych i poważnych zastosowań dostęp sekwencyjny jest niewiarygodnie wprost nieekonomiczny. Musiały więc powstać pamięci o innym sposobie dostępu. I oczywiście powstały, doskonale znanym przykładem jest pamięć na dyskach elastycznych (dyskietkach). Dają one tzw. bezpośredni dostęp do danych — w uproszczeniu można powiedzieć, że czas dostępu do każdego fragmentu danych jest taki sam, nie zależy od położenia tegoż fragmentu[x]. Przykładem takiej pamięci jest również pamięć operacyjna komputera.

NOWE MOŻLIWOŚCI — NOWE KŁOPOTY

Do budowy nowych pamięci wykorzystywano walce (pamięć bębnowa) lub dyski pokryte substancją magnetyczną. Żeby na takim nośniku zapisać swoje dane muszę wiedzieć gdzie (pod jakimi adresami[xx]) jest jeszcze wolne miejsce, żeby zapisane dane móc później odczytać muszę zapamiętywać adresy podczas zapisu. Ponieważ grzebanie się w fizycznych adresach pamięci dyskowej jest dla człowieka żmudne i mało efektywne, więc wszyscy już chyba zgadują, że system operacyjny dostaje nową robotę. Do tej pory miał tylko pilnować maszyny, teraz będzie musiał również opiekować się naszymi danymi.

Żeby jakiekolwiek zadanie mogło być wykonane, trzeba je najpierw przekazać wykonawcy. Jak mogło by to wyglądać w tym przypadku. Może: „odczytaj z dysku program, który ostatnio zapisaliśmy w lewym górnym rogu". Pomysł chyba nie rewelacyjny, bo nie wszystkie dyski mają rogi, poza tym naszym zamiarem było zwalić na system zajmowanie się szczegółami technicznymi, a więc nic nie chcemy wiedzieć o rozmieszczaniu danych. Więc może „odczytaj mi ten program, cośmy zapisali w zeszły czwartek". Też nie najlepiej, bo to i długo, i w czwartek zapisaliśmy 4 programy, więc trzeba by podawać również dokładną godzinę zapisu. Zdecydowanie potrzebne jest nam coś innego.

Rozwiązanie jest następujące: umawiamy się, że informacje będą zawsze przekazywane porcjami. Każdej porcji nadamy nazwę ... i niepowtarzalną, czyli nazwa ta stanie się jednoznacznym identyfikatorem tej porcji informacji. Dzięki temu podając tę nazwę ... ślamy systemowi operacyjnemu o co nam chodzi. Oczywiście musi on prowadzić aktualny spis wszystkich porcji, które ma na ... dyskach. Spis ten zwykle nazywamy kartoteką (ang. directory), zaś na porcje informacji mówimy zbiór lub plik (ang. file).

Minimum informacji, które musi znaleźć się w kartotece, to nazwa zbioru, oraz adresy fizyczne, pod którymi zapisana jest zawartość (treść) zbioru. Dodatkowo można tam spotkać np. datę i godzinę założenia zbioru, jego typ, itp.

JEGO WYSOKOŚĆ ZBIÓR

Teraz uwaga, koncepcja zbioru jest podstawą praktycznej eksploatacji komputerów do dnia dzisiejszego. Pozwala przejść ze świata pojęć, na których operuje człowiek, do świata zer i jedynek maszyny w sposób w miarę bezbolesny. Dla nas jednostką informacji jest np. tekst programu, dla maszyny taką jednostką jest fizyczny sektor na dyskietce. Polecamy systemowi zapisać program w zbiorze o dowolnej nazwie np. PROGR, a on sam wybiera i przydziela odpowiednie, wolne fragmenty dysku, zapisuje na nich znaki tworzące program i zapamiętuje odpowiednie dane w kartotece. Tak to wygląda technicznie, w praktyce zaś nie pamiętamy o tych szczegółach technicznych — wydajemy polecenie zapisania i koniec, mamy zbiór, powiedzmy o nazwie PROGR. Jeśli teraz zechcemy przepisać program na inną dyskietkę, wydamy polecenie „skopiuj zbiór", jeśli zechcemy się go pozbyć, wydamy polecenie „usuń zbiór". Powtarzam raz jeszcze: zbiór jest w tej chwili podstawowym pojęciem podczas współpracy człowieka z maszyną.

Być może już zauważyliście, że wiję się jak piskorz, aby zamiast słowa dane używać słowa informacja. Powód jest bardzo prosty: w zbiorze możemy umieścić wszystko: program źródłowy, jego przetłumaczoną wersję binarną, dane, wyniki, tekst stworzony pod edytorem itd. Dzięki temu np. kopiowanie z dyskietki na dyskietkę, odbywa się zawsze taką samą komendą, czyli „kopiuj zbiór", niezależnie od tego jaka treść jest kopiowana.

Zatrzymajmy się jeszcze przez chwilę przy terminologii. Angielskie słowo file było tłumaczone na polski na wiele sposobów, ostatecznie największą popularność zdobyły: zbiór (nie mylić z pojęciem zbioru występującym w matematyce, ani z występującym w języku Pascal zbiorem elementów (ang. set)), oraz plik (nie jest to oczywiście plik czegoś, np. papierów). Można spotkać się z tłumaczeniem „file" na „kartoteka"[xxx], wtedy tylko z kontekstu możemy zorientować się, że chodzi o zbiór, a nie o kartotekę zbiorów zapisanych na urządzeniu.

Często zdarza się tak, że kartoteki zbiorów, którymi zarządza system operacyjny są bardzo rozbudowane. Także funkcji wykonywanych na zbiorach jest dużo więcej niż zdążyłem zasygnalizować. Dlatego też, często mówimy nie o pojedynczych zbiorach lecz o systemie plików (zbiorów). Jednak jego szczegółowe omówienie musimy odłożyć na inną okazję, gdyż mamy śledzić dalsze losy systemu operacyjnego.

Oczywiście jest, że wzrost wymagań musiał spowodować rozbudowę systemu, a więc także wzrost jego komplikacji i ilości zajmowanego miejsca.

Tak się jednak szczęśliwie złożyło, że dzięki tym samym urządzeniom, które spowodowały wzrost wymagań, system zyskał możliwość rozbudowy ... dla sprostania tym wymaganiom. Może trzymać część swoich rzadziej używanych, podprogramów (część siebie) na dysku czy bębnie, i w razie potrzeby szybko ściągać je do pamięci operacyjnej.

Więc znowu było doskonale, ale już wkrótce błogi spokój zaczęły przerywać niemiłe zgrzyty.

TAŃCOWAŁY DWA MICHAŁY

Pamiętamy wszyscy, że jak ten duży zaczął krążyć, to ten mały nie mógł zdążyć. Procesor komputera osiągnął prędkości wielokrotnie przewyższające prędkości urządzeń służących do wprowadzania i wyprowadzania danych (tzw. urządzeń wejścia/wyjścia). Przyczyna była bardzo prosta: w dziedzinie elektroniki obserwujemy szalony postęp, stałe zwiększanie szybkości pracy układów (oczywiście także ich miniaturyzację). Natomiast w układach mechanicznych pewne progi ogromnie trudno przekroczyć. A czytniki czy drukarki, to przede wszystkim mechanika.

Co można zrobić w takiej sytuacji? Procesor powinien komunikować się z urządzeniami, które są możliwie jak najszybsze, a więc np. z pamięciami dyskowymi. Z nich pobierać dane, na nich zapisywać wyniki. Natomiast dane na dysku muszą być przygotowane zawczasu. W uproszczeniu może to wyglądać tak: wszystko co trzeba wczytać z kart dla policzenia jednego zadania jest wczytane i umieszczone w zbiorze dyskowym. Procesor rozpoczyna obróbkę zadania odwołując się tylko do dysku, a wyniki zamiast na drukarkę wysyła też na dysk. Takie podejście stwarza jednak nowe problemy, którymi zajmiemy się za miesiąc.

Andrzej Pilaszek

x) W przypadku taśmy najbardziej nam przeszkadza właśnie to, że jak byśmy jej nie ustawili, to zawsze któreś dane będą daleko — trzeba będzie na nie długo czekać.

xx) Cała powierzchnia dysku podzielona jest na fragmenty, aby można było się do tych fragmentów odwoływać każdy z nich musi mieć jakiś jednoznaczny identyfikator, pozwalający odróżnić go od innych. Tym identyfikatorem jest adres fizyczny fragmentu, określający jego dokładne położenie na nośniku.

xxx) Słowo „file" w swoim pierwotnym, używanym w tradycyjnym biurze, znaczeniu to właśnie kartoteka dokumentów

UWAGA BŁĄD!

Jak zapewne dostrzegli uważni Czytelnicy artykułu „Jak zrobić majątek" (Bajtek 11/87), wszystkie zamieszczone w tym artykule fragmenty programów liczących objętość kuli są BŁĘDNE!!, gdyż dokonują obliczeń wg niepoprawnego wzoru. Oczywiście poprawny wzór na objętość kuli o promieniu R brzmi:

$$V = \frac{4}{3} \cdot \pi R^3$$

Niby niewielki błąd we wzorze dyskwalifikuje cały program, może więc ten przypadek zachęci Was do lektury artykułu, w całości poświęconego zagadnieniom poprawności programów, który ukaże się zaraz po ostatnim odcinku „Systemu Operacyjnego".

(a.p.)

CARTRIDGE

Cartridge to najczęściej małe plastikowe pudełko wyposażone w łącze, które można po prostu „wcisnąć” w komputer. Po takim zabiegu nasz znajomy komputer zaczyna się zachowywać zupełnie obco, lub przynajmniej nabiera nowych cech, o które go do tej pory zupełnie nie podejrzewaliśmy. Nie trzeba dodawać, że są to zmiany na lepsze.

W takim razie, co to jest?

Odpowiedź jest prosta. Cartridge, to po prostu dodatkowa pamięć (najczęściej jest to EPROM 2732, 2764 lub 27128) zawierająca nowy program dla komputera, lub rozszerzenia językowe (np. BASIC-a). Tutaj powinien nastąpić sprzeciw tych, którzy czytali poprzednie odcinki cyklu „PODSTAWY”. Jak to dodatkowa pamięć? Przecież większość komputerów posiada już całą pamięć jaką może „obsłużyć” mikroprocesor.

Najłatwiej zrozumieć sposób instalacji cartridge'a w komputerach, w których przestrzeni adresowej pozostawione są „dziury” w adresacji. Cartridge po prostu wypełnia sobą te „puste” adresy uzupełniając w ten sposób oprogramowanie podstawowe komputera (rys. 1). Są to przeważnie komputery szesnastobitowe o szerokiej magistrali adresowej (20 bitów lub więcej). Taki sposób jest zastosowany np. w komputerach Sinclair QL i w komputerach klasy IBM PC. W tych ostatnich jako cartridge można traktować każdą dodatkową kartę rozszerzającą, wkładaną w jedno z kilku miejsc przygotowanych do tego celu.

W komputerach ośmiobitowych, rzecz jest trochę bardziej skomplikowana, nie ma miejsca na dodatkową pamięć. Omówię tu dwie metody, które mogą mieć zastosowanie.

Oprócz przestrzeni adresowej przeznaczonej na pamięć komputera niektóre procesory (INTEL, ZILOG) posiadają drugą, przeznaczoną na urządzenia wejścia/wyjścia. Nie może być ona wykorzystana jako przestrzeń dla pamięci programu, ale można zastosować mechanizm przepisywania zawartości cartridge'a do obszaru normalnej pamięci (RAM) komputera, gdy zostanie stwierdzone, że jest on wetknięty do komputera. Sprawdzenie takie odbywa się najczęściej zaraz po włączeniu komputera do sieci poprzez odczytanie konkretnej komórki pamięci, a w tym przypadku konkretnego „Portu wej/wyj”. Jeżeli cartridge jest na swoim miejscu można uruchomić procedurę przepisującą jego zawartość do pamięci RAM. Taki sposób odczytywania cartridge'a powoduje, że nawet gdyby się go nam udało wetknąć w działający już komputer bez spowodowania uszkodzenia, to i tak dodatkowe funkcje nie będą obecne bo komputer nawet nie będzie o tym wiedział.



Cartridge do Commodore 64. W celu zabezpieczenia przed kopiowaniem nie zastosowano typowego EPROM-u a układ scalony napylony jest bezpośrednio na płytkę.



O wiele bardziej popularna metoda to wbudowanie w komputer mechanizmu, który umożliwi „wycięcie” kawałka istniejącej pamięci komputera i w tak stworzoną lukę umieści tę dodatkową pamięć z modułu. Metoda ta jest bardzo prosta (przez co i tania), nie rezerwuje żadnych zasobów komputera (np. konkretnych portów wej/wyj) do wyłącznych celów obsługi cartridge'y i, co może najważniejsze, umożliwia podmienianie również pamięci stałej (ROM) komputera. Ta ostatnia możliwość daje szansę, aby komputer po włączeniu zgłosił się innym językiem programowania.

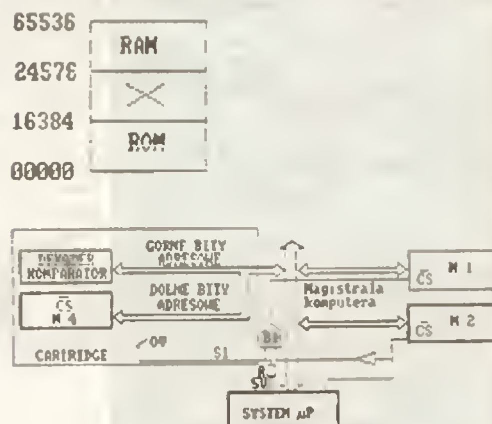
Taki cartridge jest bardzo prostym urządzeniem. Koszt jego wykonania we własnym zakresie nie przekracza zwykłe ceny prostych interface'ów. Decydująca jest w tym przypadku cena pamięci EPROM rzędu 4—8 tys. zł.

Teraz bliżej o budowie samego cartridge'a i o jego dziwnych właściwościach usuwania sobie z drogi „niepotrzebnych” obszarów pamięci.

Pamięci komputerowe są tak zbudowane, że istnieje wybrany sygnał, który umożliwia odczyt, lub zapis wybranego bajtu. Najczęściej sygnał ten nazywa się 'CS' (ang. 'chip seleckt' — wybór układu) lub 'CE' (ang. 'chip enable' — uaktywnienie układu). Mikroprocesor adresujący konkretny bajt musi wpierw uaktywnić układ pamięci, w którym się on znajduje. Jeżeli tego nie zrobi, lub jeśli mu się w tym przeszkodzi, będzie tak, jakby tej pamięci w ogóle nie było. Jest zatem sposób aby unieruchamiać pamięć komputera.

Komputer po włączeniu do sieci bez cartridge'a ma aktywne wszystkie „kości” pamięci. Jeżeli cartridge jest obecny, musi dezaktywować odpowiedni obszar pamięci wewnętrznej, czyli wpływać na sygnał wyboru tych układów. Musi również wtedy sam być aktywny, no i swoją adresacją odpowiadać „dziurze” którą sam stworzył.

Rys. 2 ilustruje jak może być to rozwiązane. Układy pamięci wybierane są przeważnie niskim sygnałem logicznym, tzn. 0V. Cartridge musi zmienić ten sygnał na jedynkę logiczną, tzn. 5 V. Realizuje to bramka B1 będąca integralną częścią komputera. Tak naprawdę, to jest to cały mechanizm umożliwiający podłączanie cartridge'y. Niewiele to i tanio. Prawda? Gdy cartridge nie jest podłączony, wyjście B1 jest w stanie niskim, aktywując pamięć M1. Gdy cartridge jest na swoim miejscu, wejście 2 bramki B1 jest wtedy zwarte do masy, a wyjście w stanie jeden, czyli M1 jest nieaktywne. Pierwsza część zdania wykonana. Drugim układem scalonym w cartridge'a jest najczęściej dekoder (komparator) adresu. Pamięć modułu dodatkowego jest zazwyczaj niewielka 4kB, 8kB rzadko 16kB. Do jego zaadresowania wystarczy odpowiednio 12, 13 lub 14 dolnych bitów adresowych. Pozostałe, górne bity służą do odpowiedniego umiejscowienia go w przestrzeni adresowej komputera, tzn. wpasowania w utworzoną „dziurę”. Do tego celu służy właśnie dekoder komparator. Gdy górne bity adresowe komputera będą miały określony stan wyjście dekodera/komparatora cartridge'a przyjmie wartość zero, uruchamiając tym samym pamięć M4 modułu.



Rys. 1 Przestrzeń adresowa komputera z pozostawionym wolnym miejscem 16384-24576.

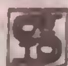
Rys. 2 Schemat blokowy podłączenia cartridge'a z odłączeniem pamięci wewnętrznej.

Dwa układy scalone, jakiś rezystor, kilka kondensatorów w odpowiednich miejscach by tłumić przypadkowe przepięcia i to jest cały cartridge. Oczywiście wszystko można komplikować. Wyobraźmy sobie, że mamy wpływ na sygnał S1, możemy go dowolnie ustawiać. Oznacza to, że możemy odłączać i dołączać moduł w czasie pracy komputera przy pomocy odpowiednich procedur programowych, czyli korzystać zarówno z nowych funkcji jak i z odłączonych zasobów komputera. Prawda, że nie w tym samym momencie, no ale można się z tym jakoś pogodzić, przecież zyskujemy o wiele więcej.

Pozostaje jeszcze jeden problem. Co jest w pamięci cartridge'a? Możemy sami napisać odpowiednie procedury i zaprogramować EPROM. Trzeba to robić zgodnie z wymaganiami dla danego komputera, podanymi w odpowiedniej dokumentacji, bardziej już profesjonalnej ale przeważnie ogólnie dostępnej. Można również przekopiować cartridge fabryczny, ale to już nie jest zupełnie legalne.

Krzysztof Czernek

WARTO PRZECZYTAĆ



Stołeczny Ośrodek Elektronicznej Techniki Obliczeniowej

INFORMATYKA mikrokomputerowa

Klaudiusz Dybowski

COMMODORE - BASIC (C-64, C-128)

COMMODORE

Warszawa 1987

COMMODORE BASIC

Stołeczny Ośrodek Elektronicznej Techniki Obliczeniowej wydał kolejną książkę z serii „Informatyka mikrokomputerowa", która w całości poświęcona jest Commodore C-64 i C-128. Jej autora, Klaudiusza Dybowskiego, nie trzeba przedstawiać czytelnikom „Bajtka", ponieważ od dawna redaguje rubrykę „Klan Commodore" oraz publikuje liczne interesujące artykuły.

„Commodore Basic" nie jest wyłącznie podręcznikiem programowania w BASICU. Jest to książka, która może pełnić rolę poradnika stale towarzyszącego użytkownikowi Commodore. Autor dzieli się w niej z czytelnikami ogromną wiedzą praktyczną i teoretyczną zdobytą podczas kilkuletniej pracy z komputerem. Uczy on, w jaki sposób należy posługiwać się sprzętem, by ustrzec się awarii, przekazuje podstawowe informacje niezbędne do poznania klawiatury, trybów pracy, edycji programów oraz objaśnia elementarne pojęcia informatyczne. Jeden z rozdziałów zawiera krótki wykład Basica V 2.0 zaimplementowanego w Commodore 64. Opisane są w nim w kolejności alfabetycznej wszystkie rozkazy, funkcje i instrukcje. Każdemu hasłu, oprócz opisu składni i działania towarzyszy krótki przykład oraz informacja o najczęściej występujących błędach. W odszukaniu interesujących nas wiadomości pomaga znajdujący się na początku książki spis rozkazów. W podobny sposób opracowano rozdział poświęcony Commodore 128. Umieszczenie w jednej publikacji materiałów dotyczących C 64 i C 128 daje znakomitą okazję ich porównania i pozwala zapoznać się z ciekawymi rozwiązaniami technicznymi stosowanymi w wersji 128 komputera.

Dla czytelników książki jedną z największych atrakcji będzie na pewno rozdział odsłaniający sekrety programowania ruchomych duszków i prezentujący możliwości graficzne Commodore. Podano tu wszystkie niezbędne informacje, które umożliwią użytkownikowi włączenie do własnych programów efektów uzyskiwanych przy wykorzystaniu układów VIC i SID.

Autor pamiętał także o tych, którzy zdecydowali się na zakupienie stacji dysków. Do tej grupy użytkowników skierowane jest kilkanaście stron omawiających współpracę z urządzeniami zewnętrznymi oraz sposób ich obsługi.

Książkę zamykają dodatki zawierające między innymi komunikaty błędów, opis portów, znaki ASCI skrócone omówienie rozkazów BASICA. Na uwagę zasługują tu rzadko publikowane rysunki znaków pojawiających się na ekranie po otwarciu cudzysłowu i wciśnięciu HOME, SHIFT lub CTRL.

Książka „Commodore Basic" na pewno nie zastąpi podręcznika programowania, ale za to czytelnik ma możliwość wszechstronnego poznania komputera. Będzie ona bardzo pożyteczna dla wszystkich, którzy mają lub chcą mieć Commodore.

(jj)

Klaudiusz Dybowski — „Commodore — Basic (-64, C-128)" Warszawa 1987, SOETO, Wydanie I. Nakład 3000 egz. Cena 960 zł.

TYLKO DLA PRZEDSZKOLAKÓW

POCIĄG

Cześć Maluchy!

Stoi na stacji lokomotywa... a za nią wagony towarowe, które trzeba załadować czekającymi w magazynach skrzyniami. Maszynista, Kubuś Literka siedzi w parowozie i czeka, aż wszystkie wagony zostaną zapełnione.

Pociąg składa się zawsze z dwóch wagonów ale mogą one być bardzo różne — długie i krótkie, wysokie i niskie. Za każdym razem trzeba więc dokładnie policzyć ile pakunków pomieszczą. Skrzynie zgromadzone są w pięciu magazynach. W pierwszym jest tylko jedna, w drugim i trzecim po dwie, w czwartym pięć a w szóstym aż dziesięć skrzyń. Jest jednak pewna trudność. Załadowując pociąg musimy opróżniać całe magazyny. Na przykład z magazynu piątego trzeba wziąć 10 skrzynek i ani jednej mniej. Oczywiście możemy też — jeśli tak nam będzie pasowało — nie brać nic z tego magazynu.

Jeżeli obliczymy, że w jednym wagoniku zmieszczą się cztery skrzynki a w drugim trzy, to będziemy się starali tak wybrać magazyny, aby w sumie było w nich siedem skrzyń. W tym przypadku będą to magazyny drugi lub trzeci (2 skrzynki) oraz czwarty (5 skrzyń).

Gdy prawidłowo wybierzemy magazyny, to znaczy liczba skrzyń znajdujących się w nich będzie dokładnie równa liczbie miejsc na wagonach, wówczas pociąg z ładunkiem odjedzie, a na peronie pojawi się następny skład. Jeśli jednak skrzynek będzie zbyt dużo, pociąg nie ruszy się z miejsca a my — za karę — dostaniemy inne wagony do załadunku.

Zastanówmy się teraz, czy może się zdarzyć, że nie będziemy mogli załadować pociągu? Największy wagon może pomieścić 9 skrzyń a więc w dwóch takich wagonach należy umieścić 18 pakunków. To największa ich liczba jaka może w naszej zabawie wystąpić. We wszystkich magazynach mamy w sumie 20 skrzynek a więc możemy być spokojni, że towaru nie zabraknie. Czy jednak każdą liczbę (nie większą niż 18) będziemy mogli przedstawić jako sumę liczb pakunków w poszczególnych magazynach? Przerwijcie na chwilę czytanie i spróbujcie to sprawdzić.

No i co?

Znaleźliście może taką liczbę, której nie da się uzyskać? Jeśli tak, to radzę sprawdzić jeszcze raz bo to z pewnością pomyłka. Liczby skrzyń w magazynach są tak dobrane, że można z nich ułożyć dowolną liczbę od 1 do 20.

Spróbujcie sobie teraz przypomnieć, czy spotkaliście się już z podobnym problemem. A kto z Was bawił się wagą z szalkami i odważnikami? Pamiętacie jakie odważniki są w komplecie? No właśnie, np. 10 dkg, 20 dkg, 20 dkg, 50 dkg, 1 kg, 2 kg, 2 kg,... itd. Dzięki temu możemy ważyć z dokładnością do najmniejszego odważnika w zestawie.

Widzicie więc, że załadunek pociągu może mieć wiele wspólnego z wagą i... matematyką.

• • •

Program napisany jest na komputerze Amstrad. Pod wydrukiem podane są linie, które trzeba zmienić aby można go było uruchomić na Commodore i Spectrum. Niestety, w przypadku Atari konieczna jest zmiana konstrukcji całego programu, gdyż ten komputer nie posiada możliwości korzystania z tablic tekstowych.

Romek

```
                             ---   T
!    !   !     !  !  !---
o---o  -o---o-  O-oooo\

                             ---   T
!    !   !     !  !  !---
o---o  -o---o-  O-oooo\

                       ---   T
!   !   !   !   !  !---
o--o  o--o  O-oooo\

                                 ---   T
!    !
!    !   !       !  !  !---
o---o  -oo---oo-  O-oooo\

                          ---   T
!     !   !   !   !  !---
-o---o-  o--o  O-oooo\

!###!                   ---   T
!###!  !##!   !;!---
o---o  o--o  O-oooo\

            !###!    ---   T
!#######!  !###!  !;!---
-oo---oo-  o---o  O-oooo\
```

```
10 REM ################################
20 REM ########### Na kozie ###########
30 REM ################################
98 REM
99 REM ###### dane poczatkowe ######
100 CLS
110 LET w=0:LET s=0
120 LET d=12
130 DIM w$(5,5)
140 FOR i=0 TO 5:FOR j=0 TO 5
150   READ w$(j,i)
160 NEXT j:NEXT i
170 DATA "---  T ",",",",","
   ",",",","
180 DATA "  !---  ",",",",","
   ",",",",!",,"
190 DATA "O-oooo\","o--o","o---o","-o---
o-","o---o","-oo---oo-"
200 DATA "---  T ",",",","
   "," ###",","
210 DATA ";!--- "," ## "," ###!"," ####
# "," ###' "," #######!"
220 DATA "O oooo\","o--o","o---o","-o---
o-","o---o","-oo---oo-"
230 DIM lad(5)
240  FOR i=1 TO 5
250   READ lad(i)
260  NEXT i
270 DATA 2,3,5,6,7
280 DIM magazyn(5)
290  FOR i=1 TO 5
300   READ magazyn(i)
310  NEXT i
320 DATA 1,2,2,5,10
330 LET los1=INT(RND(1)*5)+1
340 LET los2=INT(RND(1)*5)+1
999 REM ######### gra #########
1000 LOCATE 1,1
1010 IF s=lad(los1)+lad(los2) THEN LET
w=3
1020  FOR i=0 TO 2
1030   PRINT w$(los1,i+w);" "w$(los2,i+
w);" "w$(0,i+w)
1040  NEXT i
1050 PRINT:IF s>lad(los1)+lad(los2) THEN
 PRINT "Za duzo":FOR t=1 TO 4000:NEXT t:
RUN
1060 PRINT
1070  FOR i=1 TO 5
1080   PRINT i;
1090    FOR j=1 TO magazyn(i)
1100     PRINT "#";
1110    NEXT j
1120   PRINT "          "
1130  NEXT i
1140 IF w=3 THEN PRINT:PRINT "Wspaniale
   ":PRINT"              ":GOTO 5000
1150 PRINT: PRINT "Ktory magazyn";
1160 INPUT a
1170 IF a<1 OR a>5 OR a<>INT(a) THEN
PRINT: PRINT "Oszukujesz!     ":FOR t=1
TO 4000:NEXT t:RUN
1180 LET s=s+magazyn(a)
1190 LET magazyn(a)=0
1200 GOTO 1000
4999 REM ##### odjazd pociagu #####
5000  FOR p=1 TO d
5010   LOCATE 1,1
5020     FOR t=1 TO 50*(d-p):NEXT t
5030    FOR i=3 TO 5
5040     FOR r=1 TO p
5050      PRINT " ";
5060     NEXT r
5070    PRINT w$(los1,i);" "w$(los2,i);
" "w$(0,i)
5080    NEXT i
5090  NEXT p
5100 RUN
```

```
1 REM ##### Commodore 64 i 128 #####
100 PRINT CHR$(147);
1000 PRINT CHR$(19);
5010   PRINT CHR$(19);

1 REM ############# Spectrum ###########
##
330 LET los1=INT(RND*5)+1
340 LET los2=INT(RND*5)+1
1000 PRINT AT 0,0;
5010   PRINT AT 0,0;
```

CHWALĘ „BAJTKA”

Rozmowa z prof. Horstem Tschoppe z Uniwersytetu Technicznego w Dreźnie, dyrektorem Centrum Informatyki

— Panie profesorze, czy uważa Pan, że społeczeństwo NRD jest już przygotowane na nadejście ery informatycznej?

— Jednoznacznie tak. NRD bardzo wcześnie rozpoczęło własną produkcję elementów mikroelektronicznych, kombinat „Carl Zeiss” w Jenie i zakłady w Erfurcie należą do najbardziej priorytetowych inwestycji. Wysoki poziom przemysłu podzespołowego stanowi gwarancję rozwoju także branży komputerowej.

W tej właśnie branży rozbudowaliśmy ostatnio znacznie kombinat „Robotron” głównego producenta techniki komputerowej. Rozwijamy także współpracę produkcyjną z innymi krajami, głównie ze Związkiem Radzieckim.

Obok bazy materiałowej i produkcyjnej decydującą rolę dla rozwoju informatyki ma kształcenie. Mamy długofalową koncepcję. Zakłada ona z jednej strony kształcenie na poziomie wyższym w politechnikach, uniwersytetach a z drugiej także w szkołach średnich. To wszystko skłania mnie do pozytywnej odpowiedzi na to pytanie.

— Jaką rolę w edukacji informatycznej grają w Pana kraju małe systemy — komputery osobiste i domowe?

— Nasz potencjał produkcyjny zorientowany jest dziś głównie na produkcję komputerów osobistych i biurowych. Chodzi nam głównie o wprowadzenie tej techniki do przemysłu i całej gospodarki. W dalszej kolejności widzimy konieczność tworzenia sieci łączących komputery „zatrudnione” na poszczególnych stanowiskach pracy. Chcemy upowszechnić m.in. systemy CAD CAM oparte na zastosowaniu komputerów EC 1840 (NRD-owski odpowiednik IBM PC XT). Nie mamy jeszcze podstawowej bazy komputerowej, jaką tworzyć mogą komputery domowe. Dla indywidualnego użytkownika. Wychodzimy z założenia, że ważniejsza jest dziś produkcja materialna.

Tu, gdzie rozmawiamy, na Targach Mistrzów Jutra w Lipsku pokazujemy jednak także komputery domowe „Robotron” KC 8102 i podobne. Kiedy będzie można kupić je w sklepie — nie wiem, na pewno jednak w tej pięciolatce. Na razie trafiają do szkół.

— Czy wśród swoich studentów widzi Pan Profesor modę na komputery, fascynację nimi?

— Z początku z pewnością jest fascynacja. Trzeba ją odpowiednio wykorzystać. Myślę, że wraz z rozwojem zastosowań moda przekształci się w stabilne zainteresowanie zarówno wśród studentów jak i reszty młodzieży.

— A czy trudniej dostać się na informatykę niż na inny kierunek studiów?

— Tak, znacznie trudniej. W naszym centrum informatycznym immatrykulujemy rocznie 350 studentów. A mamy 1000 podań o przyjęcie.

— A ci, którzy już kończą studia informatyczne. Czy łatwo im zdobyć naprawdę ciekawą pracę?

— Bez żadnych problemów. Przemysł chciałby, byśmy dawali mu co roku 900–1000 absolwentów, trzykrotnie mniej niż możemy. W propozycjach jakie do nas przychodzą są różne, część bardzo interesująca, część taka sobie. Ale z pewnością każdy otrzymuje pracę odpowiednią do swoich kwalifikacji — jest przecież w czym wybierać.

— Można też to inaczej ująć — po prostu brakuje informatyków.

— No cóż, więcej kształcić nie możemy. Nasze możliwości są ograniczone — musimy przecież brać pod uwagę pojemność laboratoriów, sal wykładowych, domów studenckich.

— Można byłoby ograniczyć nabór na inne kierunki studiów, nie tak dziś nowoczesne.

— W naszych warunkach jest to rozwiązanie teoretyczne. Jak Pan wie przyrost naturalny w NRD nie jest wielki, nie był też wielki przed 18—20 laty. Dlatego też względy demograficzne oraz wymóg rozwoju innych dyscyplin równie ważnych dla obecnej rewolucji naukowo-technicznej, np. ochrony środowiska znacznie nas ograniczają. Proporcji w kształceniu nie możemy układać według dzisiejszej mody decydują one przecież o tym, jakimi kadrami dysponować będziemy za lat kilka i kilkanaście.

— Nie przewidujemy chyba jednak obaj zmierzchu informatyki?

— Oczywiście, że nie. Jednak co trzeba także brać pod uwagę w NRD 10—15 proc. wszystkich studentów stanowią studenci kierunków technicznych. Ich wszystkich także zapoznajemy z różnymi gałęziami wiedzy komputerowej. Przyszły inżynier budowy maszyn będzie znał nie tylko podstawy programowania lecz także będzie potrafił np. opracowywać systemy sterowania.

— Pan profesor mówi o inżynierach przyszłości, ja chcę się zapytać o tych już pracujących 30—40-letnich. Jak będą uzupełniać swoje wykształcenie? Czy wobec tak szerokiego wprowadzania systemów CAD/ /CAM wielu z nich czuje się wręcz zagrożonych.

— To bardzo skomplikowany problem zahaczający o społeczną stronę komputeryzacji. W 1990 roku, jak zakładamy, 550 tys. ludzi pracować będzie na stanowiskach CAD/CAM. Na pewno nie będą to tylko ci, którzy właśnie teraz studiują. Dlatego też kształcenie ustawiczne ma tak duże znaczenie. Mamy z przedstawicielami przemysłu wspólną koncepcję tego kształcenia. Stworzyliśmy centra doskonalenia zawodowego, gdzie zapoznajemy inżynierów z techniką komputerową. Szerokie wprowadzenie tego programu wymaga znacznej bazy materialnej i technicznej.

Zostaną oczywiście i tacy, którzy nie dadzą się „skomputeryzować” — z konserwatyzmu, z lenistwa, czy ze strachu przed czymś nieznanym. Trzeba im będzie także stworzyć miejsca pracy — to kwestia polityki socjalnej.

— Dużo ich będzie?

— Niewielu. Wprowadziliśmy dla przykładu na kolei komputerowy system rezerwacji i sprzedaży biletów. Z początku mieliśmy wielkie trudności z przeszkoleniem personelu który tak przyzwyczaił się do starych metod. Ale w końcu tych, którzy musieli zmienić pracę było tylko kilka procent.

— Wiem, że ma Pan wśród swoich studentów także Polaków.

Jak Pan ocenia ich wiedzę, przygotowanie do podjęcia studiów informatycznych?

— Są, na ogół, znakomicie przygotowani. Mają dużo lepiej niż ich niemieccy koledzy opanowane podstawy — w szczególności matematykę. Sporo także wiedzę już o informatyce. Jest to również zasługą waszego „Bajtka”, który chwalę za przekazywanie podstaw wiedzy informatycznej. Przydałby się nam taki magazyn.

Rozmawiał
Grzegorz Onichimowski

ELEKTRONICZNY TŁUMACZ

Komputerowe wspomaganie tłumaczenia CAT powoli zyskuje coraz większą liczbę zwolenników. O jego zaletach przekonali się pracownicy jednej z hamburskich firm. Zainstalowanemu w biurze komputerowemu systemowi Logos powierzono tłumaczenie z niemieckiego na angielski instrukcji obsługi urządzeń produkowanych w tym przedsiębiorstwie. Tłumaczone są nie tylko pojedyncze słowa, ale i całe zdania. Wydajność tłumacza wzrasta dzięki temu kilkakrotnie, mimo że zredagowany tekst wymaga poprawek stylistycznych.

Utworzenie i szybki dostęp do banków danych zawierających odpowiednie słownictwo nie jest już dzisiaj największym problemem. O wiele trudniejsze jest dla komputera zrozumienie skomplikowanych reguł gramatycznych. Opracowane dotychczas programy pokonują tę przeszkodę tylko w przypadku stosunkowo prostych pod względem budowy zdań i operujących ograniczonym zasobem słownictwa tekstów fachowych.

W Kanadzie działa w pełni zautomatyzowany system tłumaczący prognozy pogody. Od 1976 roku do tłumaczenia protokołów posiedzeń EWG wykorzystywany jest program Systran. Znano go już od 1970 roku, kiedy to został użyty w amerykańskim lotnictwie do sporządzania rosyjsko-angielskich przekładów. Mimo licznych wad specjaliści posługują się do dziś jego różnymi wersjami.

Zdaniem ekspertów przyszłość komputerowego tłumaczenia należy do kompleksowych systemów, które będą w stanie zrozumieć złożone związki znaczeniowe w zdaniach. Na zlecenie EWG grupa fachowców językoznawców i informatyków łamie sobie głowy nad ambitnym projektem Eurotra, który ma przyćmić możliwości programu Systran. Ich celem jest komputerowe tłumaczenie tekstów urzędowych na dziewięć języków. Realizacja projektu przewidywana jest w 1989 roku.

(jj)

BALTCOM '87

Wystawcy byli zadowoleni, bo mogli zarobić, a organizatorzy pochwalić się udaną mimo wszystko imprezą.

— Niech pan da jedną — prosi kilkunastoletni chłopak.
— Na co ci? — pada pytanie.
— Tak sobie, do kolekcji.

Pan przy stoisku mięknie i podaje młodzieńcowi reklamówkę jednej z firm, biorących udział w pierwszej Ogólnopolskiej Giełdzie Komputerowej „BALTCOM '87”, która odbyła się w Gdańsku w dniach 17-20 listopada.

Młodzież dopisała, tłumnie waląc do Domu Technika i hali „Olivii”, zobaczyć co nowego w komputerowym interesie. Młodzież się jednak rozczarowała, bo wystawiany sprzęt do zabawy się nie nadawał. Zamiast grać np. w „Latającego niszczyciela”, młodsi ludzie patrzyli na klawiatury i monitory z obrazami firmowymi wystawców, czekając aż pani nauczycielka zarządzi odwrót. Z nudów brali więc co popadło, w postaci instrukcji i nalepek.

Giełda miała służyć robieniu interesów, toteż przeznaczona była dla poważnych ludzi. Do Gdańska przyjechało 61 firm z całej Polski, a ściśle mówiąc z Warszawy, Poznania i Trójmiasta. Organizatorzy imprezy zapewnili przyzwoite warunki do pokazania wszystkiego (co jest komputerem) na sprzedaż. Tereny wystawy były jednak ograniczone, gdyż w hali „Olivii” nie udało się rozmrozić sztucznego lodowiska. Kiedy więc targowano się o ceny i bity, hokeiści tuż obok biegali za krążkiem. Nikomu na szczęście to nie przeszkadzało.

Pierwsze dwa dni giełdy poświęcono na zawieranie kontraktów, spotkania promocyjne czy działania informacyjne. A wszystko to dla blisko 400 przedstawicieli przedsiębiorstw zainteresowanych zakupem sprzętu, oprogramowania czy współpraca w dziedzinie komputeryzacji polskiego przemysłu.

Prawie 60% wystawianego sprzętu pochodziło z importu, głównie z Tajlandii, gdzie ceny są stosunkowo niskie. Reszta urządzeń, pokazana na giełdzie, wyprodukowana została w Polsce, to znaczy w Polsce została poskładana, gdyż większość elementów także pochodzi z importu. Honoru naszej ojczyzny broniło jedynie kilka komputerów rodzimej produkcji. M.in. MAZOVIA 1016 i MAZOVIA 1016.3.TURBO.

Niewątpliwym szlagierem giełdy był nowy wyrób firmy IBM z serii PS/2 (Personal System 2) model 30 z twardym dyskiem, dyskiem 5.25" i wchodzącym (?) standardem 3.5" oraz myszką i całym pakietem efektownych programów graficznych. Ponadto efektownie prezentowały się kolorowe monitory wysokiej rozdzielczości np. Mitsubishi 1280x1024 punkty, kolorowe plotery Rolanda i oczywiście sprzęt VIDEO już nieodzowny na takich imprezach.

Obecność na „BALTCOMIE”, w większości, firm polonijnych i małych spółek potwierdziło regułę, że przedsiębiorstwom państwowym nie opłaca się handlować komputerami.

Równolegle z giełdą funkcjonował, powołany na tę okazję, Sekretariat Naukowy, który zajmował się doradztwem handlowym oraz organizacją konferencji naukowej. W ciągu czterech dni odbyło się kilkanaście wykładów związanych z problemami informatyki i elektroniki. Były to tematy czysto praktyczne, odbiegające od kanonu naukowego wykładu, co jednak okazało się wielkim plusem, zważywszy na fakt, że na giełdę zjechali głównie managerowie i inżynierowie, a nie naukowcy.

Krzysztof Czerwiński
Krzysztof A. Nowostawski

SPRZĘŻENIE ZWROTNE

Drogi Bajtku!

Na listy Czytelników odpowiada Marcin Waligórski.

Mijają właśnie dwa lata od momentu ukazania się „Bajtka" w nowej szacie wydawniczej — a zarazem dwa lata dialogu z czytelnikami na łamach niniejszej rubryki. Z tej okazji pragnę dokonac króciutkiego podsumowania.

Skierowana przez nas swego czasu do czytelnikow prośba o opatrywanie listów dopiskami (określającymi, do którego ze stałych działów „Bajtka" korespondencja się odnosi) spotkała się z życzliwym odzewem. Przez cały czas znakomicie ułatwiała i ułatwia nam to pracę. Wszystkie listy są uważnie czytane, a zawarte w nich uwagi w każdym przypadku brane pod uwagę przy redagowaniu pisma. Niestety, nie na wszystkie uwagi i prośby jesteśmy w stanie odpowiedzieć pozytywnie. Łatwiej jest nam ustosunkować się do krytyk strony merytorycznej „Bajtka". Znacznie trudniej jest uwzględnić propozycje zwiększenia objętości działów, numeru itd. — bo tu wiążą nas trudności natury technicznej i ekonomicznej.

Do działu „Sprzężenie zwrotne" trafia część kierowanych do redakcji pytań. Na pozostałe odpowiadają redaktorzy poszczególnych „Klanów".

„Sprzężenie" jest rubryką dla uważnych czytelników. Ze zrozumiałych względów przyjęto zasadę nie publikowania pytań, które już raz na łamach tej rubryki zostały postawione i objaśnione. Podobnie rzecz ma się w przypadku pytań, na które odpowiedz można znaleźć bez trudu w numerze „Bajtka". Rubryka unika w ten sposob spiętrzania się stosu spraw niezałatwionych, zaś czytelnicy — powtarzającej się lektury.

Nie wszystkie sprawy poruszane w listach mogą doczekać się wyczerpującej odpowiedzi. Najliczniejszym przykładem są tu prośby o pomoc w wyborze komputera. W ogromnej większości tych listów jest zdecydowanie zbyt mało danych, by udzielić jakiejkolwiek rady. Innym przykładem mogą być pytania o pewne rozwiązania programowe bez podania dokładnie, jaką rolę powinny spełniac i w jakim środowisku programowym pracować.

„Sprzężenie zwrotne" nie zamieszcza ogłoszeń, propozycji nawiązania kontaktów itp. W tym celu została utworzona rubryka „Indywidualny Bank Danych" oraz osobno „Klan nietypowych". Jest wszakże jeden wyjątek. Jak zapewne czytelnicy zauważyli, „Sprzężenie" od dłuższego czasu stara się pomagać niepełnosprawnym użytkownikom komputerów. Sądzę, że możemy liczyć na zrozumienie.

Bieżący odcinek „Sprzężenia zwrotnego" poświęcamy całkowicie krytykom pod adresem tej rubryki. Bardzo cieszą nas tego rodzaju listy, gdyż są one znakomitą pomocą w redagowaniu rubryki.

W numerze 10/87, w odpowiedzi na list czytelnika pytającego o dostępne programy muzyczne dla mikrokomputera ZX Spectrum, zamieszczona została zaskakująca informacja, że takie programy nie istnieją! Otóż jako muzyk oraz posiadacz tego komputera pragnę wyjaśnić, że pomimo ograniczeń sprzętowych wymienionych w odpowiedzi na list — na Spectrum istnieje szereg udanych programów muzycznych.

Zdecydowanie najlepszym jest jak dotąd „MUSIC BOX". Nie jest to w żadnym razie program prymitywny — pozwala on na zapis dwugłosowych utworow muzycznych z towarzyszeniem perkusji, której brzmienie można programować na specjalnym edytorze. Napisany utwór można odtwarzać w różnym tempie, dowolnie modyfikowac, a także przekształcic w niewielki blok kodu, działający samodzielnie bez programu głownego. Dla demonstracji do programu dołączonych jest kilka gotowych utworów opartych na melodiach zespołu „WHAM".

Innym znanym programem jest „Music Composer" firmy Romantic Robot. Program posiada dość duże możliwości w zakresie pisania i przetwarzania 1-głosowej melodii: zmiana tempa, tonacji, klucza, artykulacji dźwięku i metrum utworu.

Oba opisane programy posiadają bardzo wygodne edytory, w których część klawiatury Spectrum traktowana jest jako klawiatura fortepianu.

Inne programy muzyczne to np. Music Maker, Music (nauka podstaw zapisu nutowego), Mozart (program samodzielnie „komponujący" dwa glosy do zadanej melodii — odtworzenia przy pomocy podłączonego układu AY-3-8912).

Jeżeli chodzi o możliwości muzyczne Spectrum, to zapraszam do posłuchania (oczywiscie przy pomocy podłączonego do gniazda MIC wzmacniacza) muzyki z gier takich, jak np. Agent X czy Chronos. Muzyka w tych grach jest 5-kanałowa (!), przy pełnej niezależności każdego kanału, z perkusją i dowolnie zmieniającą się dynamiką.

Termin „muzyka" nie zakłada w żadnym razie istnienia wielu kanałow o dowolnie programowanym brzmieniu. Większośc instrumentów muzycznych jest 1-głosowa, i ma jedną, charakterystyczną barwę. Jeśli chodzi o wymienione efekty, to można je uzyskać na drodze sprzętowej i programowej (ta druga ma wiele wad, ale nie można uznac, że nie istnieje!).

Jan Kubica
ul. Prosta 14/4
53-509 Wrocław

List ten wybrałem spośród kilku, które bronią edytorów muzycznych dostępnych na komputerze Spectrum. Wnoszę z tego, że moja lakoniczna odpowiedź o wyjątkowo skąpych możliwościach muzycznych tego komputera (Bajtek nr 10/87) wywołała wiele emocji.

Spieszę zatem donieść, że nadal uważam program „MUSIC BOX" za prymitywny. Nie chodzi tu o niedostatki samego programu — możliwość tworzenia dowolnych dwudźwięków należy uznać za jego wielką zaletę. Chodzi o to, że wyrafinowanie samego programu niewiele daje użytkownikowi. Programy typu „MUSIC BOX" czy „Music Composer" pozwalają na pewne proste wprawki z melodyki (z dodaniem efektów perkusyjnych), ale np. takie dziedziny muzyki jak harmonia leżą całkowicie poza zasięgiem możliwości ich użytkowników.

Moja odpowiedź dotyczyła programów muzycznych w wąskim tego terminu znaczeniu — i tu muszę przyznać się do przeoczenia. Istnieje szereg programów, które wspomagają pracę muzyka w innych fazach niż wykonanie.

Mówi Pan o programach „Music" i „Mozart", ja ze swej strony dodać mogę choćby „Guitarist", który początkującemu gitarzyście pozwala dobrac akompaniament do zadanej melodii, ze wskazaniem odpowiednich chwytów. Takimi programami jest także przynajmniej kilka będących w obiegu symulacji metronomu.

Pragnę również dodać, że użytkownicy ZX Spectrum nie powinni wpadać w kompleksy — znajdują się bowiem w takiej sytuacji, jak właściciele „klonów" IBM PC.

Chciałbym zapewne nie tylko w swoim imieniu zapytać, czym jest spowodowana niechęc wszystkich pism komputerowych do standardu MSX.

Zajmujecie się nawet takimi komputerami, jak C-16, VIC-20, widmami komputerowymi w rodzaju Meritum i Elwro, Sharpem itp. itd. O MSX opublikowano 1 (słownie jeden) artykuł.

Piszę do Pana, bo zadziwiają mnie Pańskie odpowiedzi na pytania o standard MSX (1/87 str. 28). W momencie gdy już dawno wytwarzano komputery MSX 2 pisze Pan rzeczy co najmniej o rok przestarzałe. Dla poparcia swojej tezy podaję dane posiadanego przeze mnie komputera MSX 2 Yamaha AX-350 (proszę to porównać z Pańską odpowiedzią):

a. mikroprocesor Z 80A, zegar 3.58 MHz
b. pamięć RAM 128 KB, VRAM 128 KB
c. pamięć ROM 128 KB
d. ekran o rozdzielczosci 256 x 212 (do 256 kolorów) lub 212 x 512
e. ekran w trybie tekstowym 80 x 24, 40 x 24 lub 32 x 24 znaki
f. 32 sprite'y
g. stacja dysków 3,5 cala (720 KB po sformatowaniu)
h. rezydujący w ROM jeden z 4 programów o nazwie PAINTER pozwala tworzyć rysunki o rozdzielczości 512 x 424 w 512 kolorach.

Dodając do tego ogólnie znane możliwości dźwiękowe komputerow Yamaha i inne, nie wymienione przeze mnie zalety tego komputera jeszcze raz zapytuję, dlaczego MSX–owcy są dyskryminowani, pomimo iż CSH kupuje przecież SVI 738, a czytając Waszą rubrykę pt. „Indywidualny Bank Danych" można przecież dojść do wniosku, że nie jest nas tak mało. Nikt nas jednak nie widzi. Nie żądamy wiele, ale prawdy o MSX i choćby od czasu od czasu jakiś artykuł lub wydruk programu.

Wiesław Lisowski
ul. Zwycięzcow 45/13
03-937 Warszawa

Wspomniane przez Pana pytanie z numeru 1/87 dotyczyło standardu MSX, a nie MSX-2. Trzeba też zauważyć, ze standard określa parametry minimalne, a nie obowiązujące dla wszystkich komputerów danej klasy. Dlatego też już w momencie, gdy pisałem ową odpowiedź można było z łatwością wskazać komputery MSX z większą pamięcią RAM lub np. 64-ma „duszkami".

Zgadzam się z Panem, ze MSX-2 to propozycja ciekawa, zwłaszcza, gdy wzbogacona jest o atuty takie, jakie Pan opisuje, tzn. bardzo bogate możliwości dźwiękowe i oprogramowanie rezydujące w pamięci ROM. To, ze MSX nie jest obecny na łamach „Bajtka" (podobnie jak wymieniane przez Pana C-16, VIC-20, Sharp, i praktycznie rzecz biorąc, Meritum) wynika z porównania liczby tych komputerów w Polsce z liczbą komputerów, których klany znajdują się w naszym piśmie.

W 10-tym numerze „Bajtka" udzielił Pan odpowiedzi koledze Arkadiuszowi Sochanowi. Mógłbym tę odpowiedź rozszerzyć, a nawet sprostować.

Wydaje mi się, że rozwiązanie automatycznego uruchamiania programow nie stanowi większego problemu. Po prostu zamiast użycia do nagrywania zlecenia CSAVE należy użyć SAVE "C:. Tak nagrany program mozna wczytać z tasmy do komputera przez LOAD "C: i uruchomić go przez RUN, bądź wgrać przez RUN "C:, co go po wgraniu automatycznie uruchomi.

Jedynym mankamentem tej metody jest to, że zlecenie SAVE "C: generuje duże przerwy między zapisywanymi rekordami.

Istnieje również drugi sposób: uruchomienie z poziomu Basica niewielkiej procedury maszynowej. Podaję przykład. Aby skrocić wgrywanie programu mającego automatycznie startować rozbijemy go na dwa bloki:

1). Sama procedura maszynowa wywoływana z Basica:

```
0 DIM A$ (20): RESTORE
1 TRAP 13: I = 1
2 READ A: A$ (I,I) = CHR$ (A):
  I = I + 1: GOTO 2
13 X = USR (ADR (A$)
14 DATA 162, 253, 154, 169, 183, 72,
   169, 84, 72, 169, 4, 32, 182, 187,
   169, 255, 76, 4, 187
```

Ten programik należy nagrać na taśmie przez SAVE "C:

2). Właściwy program nagrywamy na taśmę zaraz po programie 1) przez normalne zlecenie CSAVE.

Wyjaśnienie działania: ten program (tzn. obydwa bloki) wgrywamy przez RUN "C:. Powoduje to wczytanie i uruchomienie programu 1). „Loader" ten wygeneruje procedurę maszynową, wypływaną w linii 13. Procedura ta kasuje program 1), wgrywa program właściwy i uruchamia go.

Co się tyczy programowalnosci klawisza RESET — ma Pan rację, jest to przycisk nieprogramowalny. Jednak zdanie „działanie klawisza RESET nie może byc zmienione programowo" jest nieprawdą. W normalnym stanie RESET zeruje procesor, nie naruszając jednak obszaru pamięci, w którym zawarty jest program w Basicu. Jednak po wpisaniu w komorkę o adresie 590 wartosci 1 program w Basicu zostanie skasowany (oczywiście po wcisnięciu RESET). Jest to więc zmiana działania — na dokładkę nie sprzętowa. Nie można jednak przypisać klawiszowi RESET innych funkcji.

J.W. Nabzdyk
ul. Kochanowskiego 5/13
72-420 Dziwnow 1

Stwierdza Pan w nrze 10/87, że na komputerze Atari XL/XE jest niemożliwe automatyczne uruchomienie programu bez komendy „RUN". Otóż jest to możliwe.

Pierwszy sposób ma zastosowanie w przypadku programów krótkich. Należy zapisać na kasetę program instrukcją SAVE "C:", a odtwarzać instrukcją RUN "C:". Sposób ten nie jest najlepszy ze względu na to, że zapis jest ok. 2-krotnie dłuższy.

Sposób drugi jest bardziej skomplikowany. Trzeba napisać tzw. „loader", który załaduje i uruchomi program.

```
10 REM LOADER
20 GR. O: POKE 709,0: POKE 710,
   15: POKE 712, 15
30 POSITION 5,5: ? „Loading ASL.."
40 POSITION 5,21: ? "RUN"
50 POKE 764,12: POKE 842,13:
   POSITION 0,19
60 CLOAD
```

Programu nie należy uruchamiać, lecz nagrać na taśmę instrukcją SAVE "C:". Potem kasujemy program. Teraz przykład programu właściwego:

```
10 REM PROGRAM
20 GR. 17
30 ? = 6; "aUTOSTART"
40 ? = 6; "sTANDARD"
50 ? = 6; "lOADER"
60 A = A + 1: IF A = 511 THEN A = 0
70 A = INT (A): POKE 709,A/2:
   GOTO 60
```

Program nagrywamy na kasetę zaraz za "loaderem" instrukcją CSAVE.

Drugi sposob wydaje się byc bardziej pożyteczny — gdyż dodatkowo umożliwia umieszczanie na ekranie napisow ukazujących się podczas nagrywania programu.

Sławomir Czajka
ul. Solec 115 m.34
Warszawa

Oba listy prezentują proste i interesujące rozwiązanie problemu automatycznego uruchamiania programu na Atari. Zwracam jednak uwagę, że są to rozwiązania połowiczne. Automatycznie uruchamiany program powinien bowiem rozpocząć działanie BEZ udziału użytkownika, a zatem niezależnie od tego, czy jest ładowany przy pomocy LOAD, czy RUN.

O pełnej automatyce w uruchamianiu programy możemy mówić np. w przypadku komputera Spectrum. Tu po prostu sam zapis programu na taśmie jest opatrzony informacją, czy program ma być (i od którego miejsca) automatycznie uruchomiony. Użytkownik ładuje program niezmiennie przy pomocy instrukcji LOAD, zaś resztę czynności związanych z uruchomieniem (lub nie) wykonuje system operacyjny.

Zgadzam się natomiast całkowicie z uwagami odnośnie klawisza RESET. Podana przez p. Nabzdyka modyfikacja jest programowa — a zatem nie miałem racji.

WSZYSTKO DLA WSZYSTKICH



GRY • GRY • GRY • GRY • GR

OFERUJE !

duży wybór gier komputerowych na kasetach do:
COMODORE, AMSTRAD, SPEKTRUM
oraz dyskietkach do: COMODORE 64/128

do nabycia po atrakcyjnych cenach
w sklepach BALTONA m.in. w:
WARSZAWIE, Aleje Stanów Zjednoczonych 26
GDYNI, Salon Komputerowy, Pułaskiego 8
POZNANIU, Wielka 24
SZCZECINIE, Gdańska 40
SWINOUJSCIU, Fińska 1
SOSNOWCU, Aleje Zwycięstwa 7

ZAPRASZAMY

Baltona Baltona

DORADZTWO INFORMATYCZNE

z zakresu :

- zakup i kompletowanie sprzetu komputerowego,
- eksploatacja systemow informatycznych,
- wielodostep, sieci komputerowe,

Realizowane przez ekspertow
z Politechniki Wroclawskiej

Sekcja Dzialalnosci Gospodarczej AZS
Politechnika Wroclawska tel.203700
50-370 Wrocław Wyb.Wyspiańskiego 27

K 241 1

STUDIO KOMPUTEROWE
ATARI-BAJT
ATARI ● AMSTRAD
COMMODORE ● SPECTRUM
oferuje:
programy uzytkowe, edukacyjne, gry, opisy, interfejsy do magnetofonow i piora swietlne — ATARI
tel. 20-80-34. Warszawa
katalogi gratis przy zamowieniu.

STUDIO KIJOWIANKA
AMSTRAD ● ATARI XL. XE. ST
COMMODORE 64. 128
Poleca literature i programy na kasetach i dyskach. Warszawa, ul Targowa 26. Rachunki oraz wysyłka poczta. Informacje za załaczeniem koperty i znaczka

---

AMSTRAD PC 1512 DDCM
sprzedam. 20-96-00 Gdynia.

C 64. Nowy, rewelacyjny program użytkowy „BAZA ZNAKOW"
Informacje:
Aleja Majowa 14/27
44-121 Gliwice.

INTERFEJS
•••CRI•••
Montowany na zamowienie przez firmę
IBS ELECTRONIC
umożliwia wspolpracę zwykłego magnetofonu jako pamięci kasetowej do komputerow
ATARI
zapis, odczyt, start-stop magnetofonu, gwarancja
Warszawa, tel. 34-16-06
w g. 10.00—14.00.

Serwis Komputerów
WIENCEK
poleca swoje uslugi w zakresie:
naprawy oraz oprogramowania
komputerów Commodore, IBM Atari, ZX Spectrum.
Katowice — Os. W. Witosa
ul. Ossowskiego 28/30 10 p.

ZAKŁAD ELEKTRONIKI MIKROKOMPUTEROWEJ

„TALCOMP"

31-464 Kraków, ul. Anieli Krzywon 23, tel. 11-91-22

POLECA UŻYTKOWNIKOM KOMPUTEROW ATARI:
— interfejsy umożliwiające współpracę komputerów ATARI z dowolną drukarką wyposażoną w złącze typu CENTRONIX
— cartridge z językiem programowania BASIC XL, BASIC XE, LOGO, ACTION
— cartridge z dowolnym programem wykonywane na życzenie Klienta
— rozszerzenie pamięci operacyjnej do 256 kB w komputerach ATARI 600 XL, ATARI 800XL, ATARI 65XE, oraz ATARI 130XE, zachowujące ABSOLUTNĄ wymienność oprogramowania z ATARI 130XE.

UWAGA!!! Dotychczasowe, wykonane w Zakładzie rozszerzenia pamięci komputerów, zostaną wymienione BEZPŁATNIE na rozszerzenia kompatybilne.

INFORMACJE:
telefonicznie w godz. 10-15
listownie: TYLKO za nadesłaną kopertą zwrotną.

SM 287

WSZYSTKO DLA WSZYSTKICH

refleks

NASZA OFERTA!!!



ASCOM TECHNOLOGIES
(FAR EAST) PTE LTD

PWPO-T „Refleks" Sp. z o.o. informuje,

że działa jako wyłączny przedstawiciel serwisowy na zasadzie zawartego kontraktu z ASCOM TECHNOLOGIES (FAR EAST) PTE LTD. Na zakupiony w tej firmie sprzęt wydawane jest w Polsce świadectwo jakości i udzielana jest roczna gwarancja, w czasie której funkcje gwaranta sprawuje na zasadzie wyłączności PWPOT „REFLEKS".
Sprzęt zakupiony w ASCOM po odebraniu przesyłki przez użytkownika jest testowany i sprawdzany bezpłatnie w PWPOT „Refleks" Sp. z o.o.

UŻYTKOWNIK OTRZYMUJE TYLKO DOBRY SPRZĘT!

Ponadto „Refleks" udzieli Państwu wszelkich dodatkowych informacji zarówno handlowych, jak i technicznych (katalogi, cenniki itp.).

Kontakt: **Przedsiębiorstwo Wdrażania Postępu Organizacyjno-Technicznego „Refleks" Sp. z o.o. Dział Importu, 02-051 Warszawa, ul. Glogera 1 tel. (02) 659-20-41, (02) 659-39-22 tlx 817530 ref pl.**

Wysyłkowo z firmy ASCOM TECHNOLOGIES (FAR EAST) PTE LTD otrzymacie Państwo sprzęt mikrokomputerowy wysokiej jakości i w krótkich terminach dostawy:

Oferta po atrakcyjnych cenach:

— kompletne zestawy mikrokomputerów PC/XT 6/8/10 MHz, PC/AT 8/10/12 MHz, PC/38612/16/20 MHz oraz inne, jak np. mikrokomputery przenośne i najnowsze typy profesjonalnych mikrokomputerów,
— pełny asortyment kart CSKD, wyposażenia i akcesoriów umożliwiających samodzielne zbudowanie mikrokomputera lub rozszerzenie zestawu już posiadanego (karty główne, grafiki, kontrolery, karty obsługi wejść/wyjść, kable, obudowy, klawiatury, zasilacze),
— pełny asortyment urządzeń zewnętrznych, takich jak: monitory monochromatyczne i kolorowe (szeroka gama typów o różnej rozdzielczości), pamięci taśmowe, pamięci na miękkich dyskach i napędy dysków twardych (o bardzo dużej pojemności i krótkim czasie dostępu), różne typy drukarek firm: EPSON, CITIZEN, STAR, PANASONIC, Amstrad, różne typy ploterów i digitizerów,
— **nośniki magnetyczne,**
— **inne wyposażenie w środki techniki biurowej,**
— **urządzenia i przyrządy elektroniczne,**
— **urządzenia techniki video,**
— **elementy i podzespoły elektroniczne.**

ASCOM TECHNOLOGIES/FAR EAST/PTE LTD
Republic of Singapore
45 Genting 05-02 Genting Warchouse Complex Singapore
1334 Republic of Singapore.
Przedsiębiorstwo Wdrażania Postępu Organizacyjno-Technicznego



Sp. z o.o.
K-185

CENY NA DZIEŃ 13 XII 1987

	GIEŁDA „BAJTKA" (tys. zł)	PEWEX BALTONA (USD)	RFN (śred.) (DM)
SINCLAIR			
ZX 81	50	—	39
ZX Spectrum 48 KB	110	115	110–150
ZX Spectrum Plus	160	—	180–230
ZX Spectrum 128 + 2	250	—	250
Drukarka SEIKOSHA GP 50S	90	—	99
TIMEX 2048	160	146	—
Joystick	4,5–7	—	4–9
COMMODORE			
C-64	220	219	320
C-128	300	299	480
C-128D	—	—	999
Amiga z monitorem kolorowym	1,2 mln	—	3000
Magnetofon 1531	45	48	30
Stacja dyskietek 1541	200	—	399
Stacja dyskietek 1571	240	299	460
Drukarka GP-500	200	—	149
Dyskietki 5 1/4 (średnia jakość)	0,8–1,5	3,5	0,3–1,5
ATARI			
65XE	150	125	110
130 XE	200	199	320
Stacja dyskietek 1050	230	187	350
Drukarka 1029	250	199	270–299
ATARI 520 STM st. dysk 0.5Mb	1,1 mln	998	800
AMSTRAD			
464 z monitorem monochromat.	280	—	400
6128 z monit. monochromat.	450	—	750
6128 z monitorem kolorowym	550	—	1000
Dyskietki 3	5	—	6–9
Stacja dyskietek 3 do 464	380	—	399
PC 1512	1,2–1,3 mln	—	999

Giełdy Bajtka odbywają się przy ul. Grzybowskiej 35 (Szkoła Podstawowa nr 25) w każdą sobotę i niedzielę od 14.00 do 19.00

POD CHOINKĄ

Świąteczna atmosfera udzieliła się organizatorom „Bajtkowej" giełdy. Większość stoisk udekorowano choinkami, a w miejscu tradycyjnych ozdób na zielonych gałązkach zawieszano kasety z programami i wszelkie mniejsze peryferia komputerowe oferowane do sprzedaży.

Nowości nie było, natomiast o gwiazdkową atrakcję postarał się Instytut Świata Arabskiego w Paryżu, który opracował dwujęzyczny system operacyjny łacinsko-arabski. Istnieje już edytor tekstów Arabstar 2001, a także dwujęzyczne wersje programów typu Lotus 1.2.3, dBase III, Wordstar 2000 itp. System dostarczany jest w postaci emulatora DOS na dyskietce i zmodyfikowanego kontrolera CGA. Dodatkowo dołączane są nalepki na klawisze ze znakami dla języka arabskiego.

Miejmy nadzieję, że przyszłoroczny Mikołaj przyniesie nam system, który pozwoli na obsługę rodzimego mikrokomputera z wykorzystaniem równie egzotycznych liter ś, ć, ż, ź, ę, ą.

(rel)



INDYWIDUALNY BANK DANYCH

Nazywam się **Dariusz Rubinkowski**, mam 13 lat. Posiadam mikrokomputer Commodore 116 z firmowym magnetofonem. Interesuję się grami komputerowymi oraz fantastyką, proponuję wymianę gier i literatury. Adres: 09-400 Płock, ul. Orlińskiego 3 m 20.

Nazywam się **Ignacy Zlajka**, mam 14 lat. Posiadam nietypowy mikrokomputer TJ 99/4a, mam duże trudności z nawiązaniem kontaktu z innymi posiadaczami tego mikrokomputera. Posiadam programy firmowe. Zainteresowania informatyka, język angielski. Adres: 38-520 Rymanów, ul. dr. Bieleckiego 10.

Wałach Leszek, uczeń 13 lat. Mikrokomputer ZX Spectrum+. Oprogramowanie: głównie gry. Oczekuję wymiany gier. Adres: 42-600 Tarnowskie Góry, ul. Kazimierza Wielkiego 16.

Bogdan Modrzejewski, lat 19. Mikrokomputer ZX Spectrum, monitor, magnetofon, joystik. Zainteresowania: sport, informatyka, gry komputerowe. Adres: 05-088 Brochów, woj. warszawskie.

Marcin Zatoński, uczeń 15 lat. Mikrokomputer Commodore + 4, magnetofon. Zainteresowania: koszykówka, informatyka. Proponuje wymianę oprogramowania oraz doświadczeń. Adres: ul. Poniatowskiego 10/18, 78-600 Wałcz

Piotr Wrzesiński, uczeń lat 18. Mikrokomputer SCHNEIDER CPC 464. Oprogramowanie: programy użytkowe, edukacyjne, gry. Zainteresowania: elektronika, motoryzacja. Proponuje wymianę oprogramowania. Adres: 96-122 Puszcza Mariańska 44.

Krzysztof Czaporowski, uczeń 13 lat. Mikrokomputer Commodore 16, magnetofon, joystik. Gry programy użytkowe. Proponuje wymianę oprogramowania i doświadczeń. Adres: 96-100 Skierniewice, ul. Orkana 7/10.

Mariusz Wyszomirski, 14 lat. Mikrokomputer ZX Spectrum, monitor, magnetofon, interfejs, joystik, pióro świetlne. Oprogramowanie: programy firmowe, użytkowe, graficzne i edukacyjne, gry. Zainteresowania: informatyka, elektronika, zastosowania komputerów. Proponuje wymianę oprogramowania i doświadczeń. Adres: 96-500 Sochaczew, ul. Al. 600-lecia 69/1.

Rafał Kotyla, lat 11. Posiada mikrokomputer Commodore 64, magnetofon oraz około 100 programów głównie gier. Adres: 32-620 Brzeszcze, ul. Słowackiego 9/5/9.

Tomasz Banachowicz, lat 13. Posiada mikrokomputer ATARI 800 XL. Chciałby nawiązać kontakt z innymi użytkownikami tego komputera w celu wymiany oprogramowania i doświadczeń. Adres: 81-107 Gdynia, ul. Stanisława Dąbka 63/III/10.

Piotr Kuczyński, lat 14. Posiada mikrokomputer AMSTRAD CPC 464, joystik. Oprogramowanie: gry, programy użytkowe. Proponuje wymianę literatury oraz programów. Adres: 69-100 Słubice, ul. Wojska Polskiego 165d/1.

Piotr Pierzynowski, lat 19. Posiada mikrokomputer Commodore 64 KB, system operacyjny CB/M. Proponuje wymianę doświadczeń. Adres: 72-610 Międzyzdroje, ul. Wiejska 2/18.

KOMPUTER I DZIEWCZYNY

Obecność komputera na wszelkiego rodzaju pokazach i kiermaszach nikogo już nie zadziwia. A cóż dopiero kiedy przy komputerze zasiada piękna dziewczyna.

Od niedawna modelki z Agencji Reklamowej „Promocja 2000" pełnią rolę hostess na wystawach.

Opanowanie zasad obsługi komputera nie wydaje się być sprawą trudną, skoro dziewczęta z Agencji Reklamowej pracy na komputerze np. typu „Atari" uczą się w ciągu czterech godzin! Zresztą stanowi to dla nich niezłą zabawę.

Małgorzata Pytka



SPIS TREŚCI BAJTKA 1986/1987

GRA O JUTRO
wywiady

SWEGO NIE ZNACIE

PRZED EKRANEM
NASTĘPNY KROK

NIE TYLKO KOMPUTERY

NIE BÓJ SIĘ MNIE

PODSTAWY

OPISY SPRZĘTU

TESTY

PROGRAMOWAĆ MOŻE KAŻDY

Języki

Programy

KLAN COMMODORE

SPIS TREŚCI BAJTKA 1986/1987

KLAN ATARI

KLAN SPECTRUM

KLAN AMSTRAD/SCHNEIDER

KLAN MERITUM

DLA KAŻDEGO

TYLKO DLA PRZEDSZKOLAKÓW

SAM PROGRAMUJĘ

OBOK KOMPUTERA

OPISY GIER

*** = mapa gry**

SPIS TREŚCI BAJTKA 1986/1987

JAK TO ROBIĄ INNI

TARGI

SAMI O SOBIE

WARTO PRZECZYTAĆ
recenzje

RÓŻNE

UWAGA CZYTELNICY „BAJTKA"!

Macie możliwość zamówienia rocznika '87 naszego pisma. Dwanaście numerów w twardej oprawie — koszt w zależności od liczby chętnych. Zgłoszenia przysyłajcie na adres redakcji „Bajtka" wraz z przedpłatą 1200 zł.

P.S. Jeżeli brakuje ci pojedynczych numerów, napisz do nas. Przyślemy Ci je za zaliczeniem pocztowym.

TALERZEM W NIEBO





Najnowszy produkt firmy Rockdale-aparatura, która pamięta pozycje stu dowolnie zaprogramowanych satelitów.

Zdążyliśmy się już przyzwyczaić do sympatycznej buzi pani Batista — prezenterki serwisu informacyjnego amerykańskiej sieci telewizyjnej CNN, która na naszych ekranach telewizyjnych pojawia się za sprawą Panoramy Dnia. To, iż nasza najmłodsza muza wyczarowała milionom Polaków, nie odbierającym jeszcze programu moskiewskiej telewizji, skrót dziennika „Wriemia", nie budzi żadnych emocji. Czemu jednak do formuły — coś ze Wschodu i coś z Zachodu dobrano, spośród całej gamy telewizyjnych potentatów zza Atlantyku, właśnie CNN?

Ludzie związani z naszą TV odpowiadają zazwyczaj powołując się na korzystne warunki wymiany materiałów. Kulisy jednak są zupełnie innej, technicznej natury. Rzut oka do wnętrza wydawanego w Wielkiej Brytanii magazynu „Satellite TV Europe" wyjaśnia prawie wszystko. CNN jest pierwszą siecią Stanów Zjednoczonych, które zdecydowała się na stałe, regularne przekazy satelitarne do Europy. Powieszony gdzieś na wysokości trzydziestukilku kilometrów nad Ziemią pojazd przekaźnikowo-nadawczy Intelsat VA-F11, emitujący programy w systemie PAL, umożliwił CNN stałą obecność w tych europejskich domach, które ozdobione zostały talerzami anten.

Specjalna redakcja CNN Europe przygotowuje frykasy amerykańskiej kuchni telewizyjnej podane na owych talerzach tak, aby wszystko było bardziej strawne dla mieszkańców Starego Kontynentu. W powszednie dni tygodnia w programie satelitarnym CNN przeznaczonym dla Europy dominują aktualizowane co godzinę serwisy informacyjne, notowania giełdowe, wiadomości ze świata amerykańskiego biznesu. W sobotę i niedzielę ukazują się dyskusje polityczne, sprawozdania z pokazów mody, quizy, konkursy, a wszystko to okraszone sporą porcją typowo amerykańskiego sportu. W sumie spory kawałek Nowego Świata przekazany via orbita dla Świata Starego. 24-godzinnym programem informacyjnym CNN rządzi Ted Turner. Jego hasło wywoławcze to: widzieć świat oczami Amerykanów. Po raz pierwszy można było się przekonać co to znaczy pod koniec września 1985 roku. Dziś, mimo faktu, iż przedsięwzięcie CNN obliczone było na odbiorców wcale nie masowych — spółki hotelowe goszczące Amerykanów, lokalne systemy kablowe Zachodniej Europy, centra konferencyjne itd., David Garland — przedstawiciel koncernu na Europę rezydujący w Londynie, odbiera coraz więcej indywidualnych zamówień.

W rubryce „dane" grudniowej edycji „Satellite TV Europe" znajdujemy 36 kanałów satelitarnych dostępnych mieszkańcom Zachodniej Europy. Niektóre z nich dopiero raczkują emitując nie więcej niż 2–3 godziny dziennie na przykład reklam, muzyki rozrywkowej i oczywiście filmów. Dotyczy to choćby norweskiego programu InfoFilm Video, nie znaczy to jednak, iż języki pozaangielskie są dyskryminowane. Miłośnikom języka włoskiego powinno wystarczyć około 14 godzin dziennie „jedynki" rzymskiej RAI, dzięki wspólnemu zachodnioeuropejskiemu satelicie Eutelsat-1F1, odbieranej doskonale zarówno w Neapolu, Północnej Afryce, czy Sztokholmie.

Przebiegając wzrokiem po parametrach poszczególnych sztucznych towarzyszy naszej planety ustawionych nad naszymi głowami również ku uciesze wielbicieli szklanego ekranu, łatwo dostrzeżemy, iż najbardziej oblężone różnego rodzaju kanałami telewizyjnymi są satelity serii Eutelsat i Intelsat. W Polsce upowszechniło się przekonanie, iż wystarczy zamontować talerz z urządzeniem odbiorczym, a cały świat należy do szczęśliwego posiadacza w ten sposób ustawionego systemu. Wszystko jest w porządku jeśli zdecydujemy się na łączność z jednym tylko orbitującym pojazdem. Intelsat VAF11 na przykład przekazuje na Ziemię aż 10 kanałów. Eutelsat — 1F1 również 10 innych. Wówczas operacja zmiany programów jest dziecinnie prosta. Ale cóż, kiedy kuszą inne satelity z bardziej obiecującą zawartością.

Wtedy zostaje już tylko operacja anteną, a to, zważywszy, iż talerze ustawione są na dachach, czy w ogródkach nie jest już, szczególnie w zimie, wielką przyjemnością. A co będzie, gdy wyniesione zostaną w przestrzeń następne satelity o nieznanych na razie pozycjach? Producenci najnowszych satelitarnych urządzeń odbiorczych pomyśleli i o tym. Na przykład firma Rockdale oferuje talerz z własnym mechanizmem obrotowym ustalającym jego pozycję, koderem sygnału i zdalnym sterowaniem. Na początku wystarczy wprowadzić do pamięci systemu domowego wielkości miniwieży pozycje wszystkich satelitów, których programy pragniemy oglądać. A potem już tylko wygodnie rozsiąść się w fotelu z podręcznym pilotem w ręce. Już teraz pracuje się nad instalacjami samoposzukującymi. Programowanie wstępne byłoby wtedy anachronizmem.

Rockdale podaje, iż jej aparatura jest w stanie zapamiętać i zlokalizować położenia 100 satelitów. Jeśli założymy, iż każdy będzie mógł hipotetycznie emitować tylko dwa kanały, hipotetycznie bo urządzenie, jak twierdzi firma jest perspektywiczne, to znaczy dostosowane do lawinowego rozwoju telewizji satelitarnej, możemy doznać pewnego rodzaju szoku. Ale czy się to komu podoba, czy nie, świat wkracza coraz gwałtowniej w epokę cywilizacji obrazowej. A to, że dzieje się to bez nas, może jednych denerwować, innych tylko smucić, a jeszcze innych uspokajać.

O tym, iż nie tylko nasz kraj nie nadąża za rozwojem technologii świadczy przykład niektórych lokalnych rad (councils) w Wielkiej Brytanii. W pewnych zakątkach tego kraju uważającego się za rozwinięty gospodarczo i w pełni demokratyczny wymagane są... specjalne pozwolenia władz terenowych na instalację talerzy o średnicy większej niż 60 cm. W Blackpool na przykład do podania o pozwolenie na montaż anteny dołączone muszą być zdjęcia budynku i ogródka, a także dokładne plany...

Lawinę zatrzymała na krótko katastrofa amerykańskiego promu Challenger i wstrzymanie lotów wahadłowców orbitalnych. Do niedawna bowiem, zachodni właściciele satelitów telekomunikacyjnych polegali prawie wyłącznie na usługach transportowych astronautyki USA. Obecnie niektóre konsorcja zamawiają operację wystrzelenia i ustawienia precyzyjnie na określonej orbicie własnego pojazdu, również w Chinach.

15 września 1987 r. z poligonu w Gujanie Francuskiej wystartowała ponad 50-metrowej wysokości zachodnioeuropejska rakieta Ariane. Kiedy pojazd tego typu zbudowany w przeważającej części przez Francuzów, eksplodował tuż po opuszczeniu wyrzutni w maju roku ubiegłego, prysły nadzieje Zachodnich Europejczyków na częściową choćby niezależność kosmiczną. Na to, iż własne sputniki lokowane będą nad Ziemią przez własne rakiety. Ponad rok trwały badania i ulepszanie Ariane. Wrześniowy sukces otwiera drogę do nowych startów. Już teraz mówi się o zastąpieniu Intelsatów i Eutelsatów nowymi modelami, a także o rozszerzeniu oferty programowej. Tak aby rzut talerzem w niebo nabrał nowych barw.

Wojciech Łuczak



U góry: Rakieta Ariane na wyrzutni w Gujanie Francuskiej. Pogrubiona część szczytowa jest pojemnikiem kryjącym satelitę komunikacyjnego.
Z lewej: Centrum satelitarno-przekaźnikowe sieci CNN.